

Kiểu chuỗi	Kiểu Danh sách (list)	Kiểu Từ điển (dictionary)	Kiểu Tuples
1. Khởi tạo chuỗi: Ten_bien_chuoi = “ ” Ten_bien_chuoi= “Hello” 2. Truy cập vào một ký tự trong chuỗi: Ten_Biến_Ch chuỗi[chỉ số] 3. Trích chuỗi con: Chuỗi_con = Tên_Biến_Ch chuỗi[chỉ số 1: chỉ số 2] Chuỗi_con = Tên_Biến_Ch chuỗi[: chỉ số] Chuỗi_con = Tên_Biến_Ch chuỗi[chỉ số :] Chuỗi_con = Tên_Biến_Ch chuỗi[:] 4. Chuyển đổi dữ liệu kiểu chuỗi: Tên_Biến_Ch chuỗi.title() Tên_Biến_Ch chuỗi.upper() Tên_Biến_Ch chuỗi.lower() int(Tên_Biến_Ch chuỗi_các số nguyên) float(Tên_Biến_Ch chuỗi_các số thực) 5. Bỏ khoảng trắng (space): Tên_Biến_Ch chuỗi.rstrip() Tên_Biến_Ch chuỗi.lstrip() Tên_Biến_Ch chuỗi.strip() 6. Tìm kiếm, thay thế chuỗi: Tên_Biến_Ch chuỗi.replace(chuỗi con 1, chuỗi con 2) Tên_Biến_Ch chuỗi.find(chuỗi cần tìm) Tên_Biến_Ch chuỗi.find(chuỗi cần tìm, vị trí bắt đầu tìm): 7. Kiểm tra chuỗi: Tên_Biến_Ch chuỗi.isupper() Tên_Biến_Ch chuỗi.islower() Sử dụng toán tử (in) để kiểm tra 1 chuỗi có ở trong 1 chuỗi khác 8. Trả về chiều dài của chuỗi: len(Tên_Biến_Ch chuỗi) 9. So sánh chuỗi: sử dụng các phép toán <, >, ==, !=, <=, >=	1. Khởi tạo danh sách Tên_danh_sách = [] Tên_danh_sách= list() Tên_danh_sách = [pt1, pt2, ...] 2. Truy cập vào một phần tử trong danh sách: Tên_danh_sách[chỉ số] 3. Chuyển đổi phần tử của danh sách: Tên_danh_sách[chỉ số].upper() Tên_danh_sách[chỉ số].lower() Tên_danh_sách[chỉ số].title() 4. Thêm, chèn phần tử: Tên_danh_sách.append(phần tử) Tên_danh_sách.insert(chỉ số, phần tử) 5. Xóa phần tử: Tên_danh_sách.pop(chỉ số) Tên_danh_sách.pop() Tên_danh_sách.remove(phần tử) Tên_danh_sách.clear() Lệnh del danh_sách[chỉ số] 6. Sắp xếp các phần tử: Tên_danh_sách.sort() Tên_danh_sách.sort(reverse=True) Hàm sorted(Tên_danh_sách) 7. Đảo ngược các phần tử: Tên_danh_sách.reverse(): 8. Trả về độ dài của danh sách Hàm len(Tên_danh_sách) 9. Tách chuỗi thành các phần tử của một danh sách: Tên_danh_sách = Tên_chuỗi.split() Tên_danh_sách = Tên_chuỗi.split(“,”) 10. Chuyển 1 chuỗi thành 1 danh sách các ký tự (tách ký tự): Tên_danh_sách = list(chuoi_nguồn) 11. Ghép chuỗi vào danh sách để tạo ra	1. Khởi tạo Từ điển: Tên_từ_điển = { } Tên_từ_điển = dict() Tên_từ_điển = {“khóa 1”:”gtr1, “Khóa 2”:gtr2, ...} 2. Truy cập <giá trị > của 1 phần tử: Tên_từ_điển[khóa] Hoặc Tên_từ_điển.get(khóa) Tên_từ_điển.get(khóa, Giá trị trả về) 3. Truy cập đến tất cả các phần tử : sử dụng phương thức item() theo cú pháp: Tên_Từ_Điển.items() 4. Truy cập đến <khóa> của tất cả các phần tử: dùng phương thức keys() theo cú pháp: Tên_Từ_Điển.keys() 5. Truy cập đến <giá trị> của tất cả phần tử: dùng phương thức values() Tên_Từ_Điển.values() 6. Thêm vào 1 phần tử: Tên_từ_điển[khóa_mới_cần_thêm] = giá trị 7. Cập nhật giá trị mới cho khóa: Tên_Từ_Điển[khóa] = giá trị mới 8. Xóa một phần tử: del Tên_Từ_Điển[khóa cần xóa] 9. Kiểm tra một <khóa> có trong từ điển hay không: <khóa> in Tên_Từ_Điển.keys() 10. Sắp xếp thứ tự các <khóa> trong từ điển sorted(Tên_Từ_Điển.keys()) 11. Tạo một tập <giá trị> trong từ điển sau khi loại bỏ các giá trị trùng nhau set(Tên_Từ_Điển.values()) 12. Danh sách Từ điển: Danh_Sách_Từ_Điển = [Tên_Từ_Điển_A, Tên_Từ_Điển_B, Tên_Từ_Điển_C]	1. Khởi tạo Tuples Tên_Tuples = () Tên_Tuples = (pt1, pt2, ...) 2. Các thao tác trên Danh sách có thể áp dụng đối với Tuples. Ngoại trừ: - Một khi tạo một Tuples, thì không thể thay đổi nội dung của Tuples - Không sử dụng các phương thức: sort(); append(); reverse(); ... 3. Lưu ý: - Khi sử dụng phương thức item() truy cập đến các phần tử của từ điển. Kết quả trả về 1 danh sách các Tuples. Do vậy, Sử dụng hàm sorted() để sắp xếp danh sách các Tuples. Bằng cách: + Sử dụng phương thức item() cho dữ liệu kiểu Từ điển + Tiếp đến, dùng hàm sorted() 4. Chuyển một danh sách thành một Tuples Tên_Tuples = tuple(Tên_danh_sách)

10. Các thao tác khác: <ul style="list-style-type: none">- Ghép chuỗi (Concatenating): Sử dụng ký hiệu (+).- Tăng số lần giá trị kiểu chuỗi: Sử dụng toán tử (*)- Chèn một dòng mới (xuống dòng): Sử dụng ký hiệu (\n)- Chèn dấu Tab: Sử dụng ký hiệu (\t)	một chuỗi: Tên_chuỗi.join(Tên_danh_sach) 12. Ghép 2 danh sách: danh_sach_1.extend(danh_sach_2) 13. Đếm Tần suất xuất hiện phần tử trong danh sách: Tên_danh_sách.count(phần tử): 14. Các hàm khác: Hàm max(Tên_danh_sách) Hàm min(Tên_danh_sách) 15. So sánh danh sách: sử dụng các phép toán <, >, ==, !=, <=, >=	13. Từ điển Danh sách: Tên_Từ_Điển = {Khóa 1: Danh_Sach_A, Khóa 2: Danh_Sach_B} 14. Từ điển của Từ điển: Tên_Từ_Điển = {Khóa 1: Từ_điển_A, Khóa 2: Từ_điển_B, ...}	
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--