

TÁC GIẢ: 8 SYNC

CỘNG ĐỒNG



[Kevin Nguyễn](#)



[Nhóm Chia Sẻ Công Nghệ](#)



[Nhóm BlockChain](#)



[Tiktok: 8 Sync](#)



[Youtube: 8 Sync Dev](#)



[Zalo](#)

KHÓA HỌC:



[Fullstack Python](#)



[Fullstack Nextjs](#)



[Fullstack Android-IOS](#)

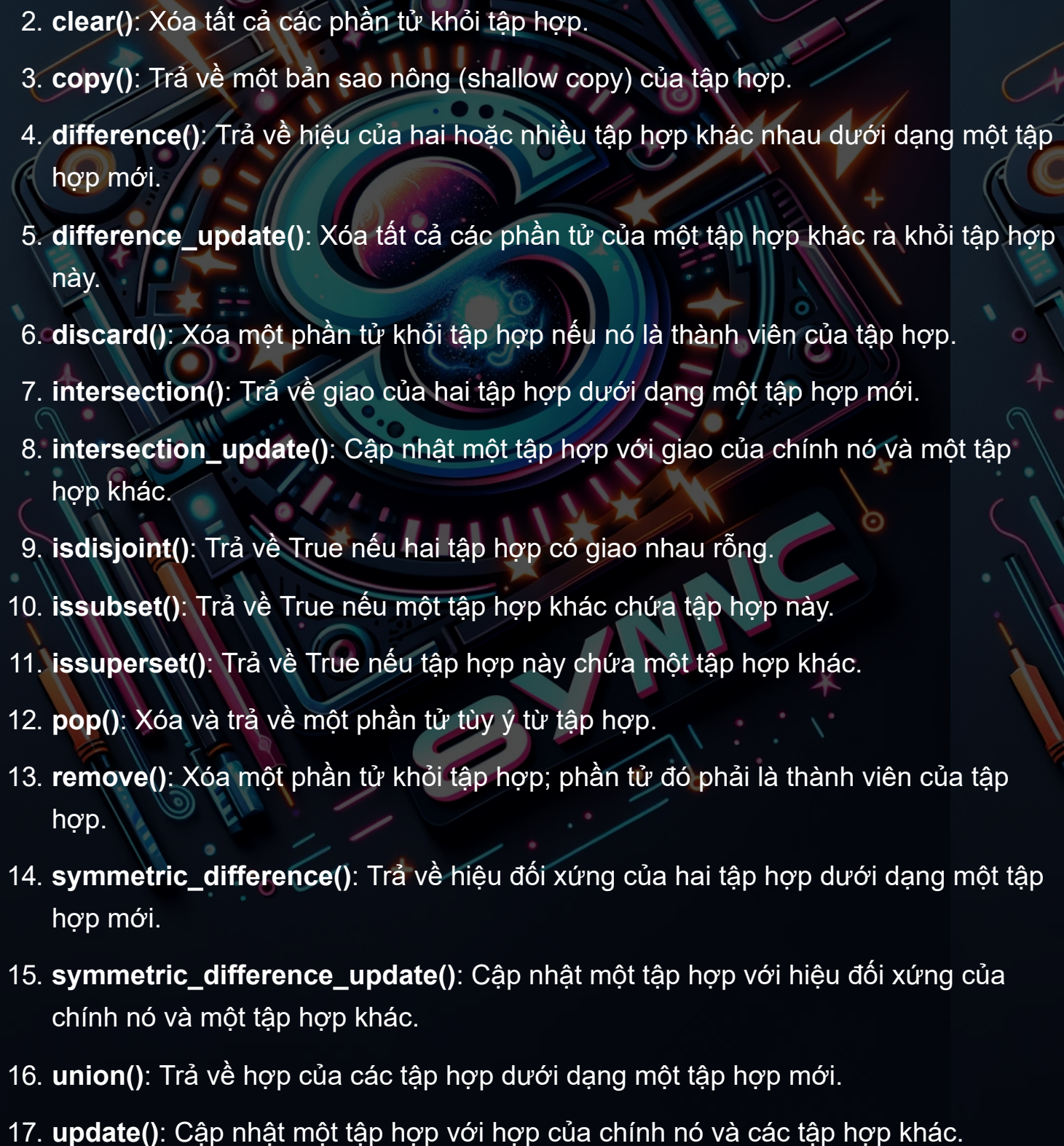
Tài liệu sẽ được cập nhật định kì và thông báo trong group nên các bạn chú ý nhen .



PYTHON - CÁC PHƯƠNG THỨC CỦA TẬP HỢP

Trong lớp tập hợp của Python, các phương thức sau được định nghĩa:

1. **add()**: Thêm một phần tử vào tập hợp.

- 
2. **clear()**: Xóa tất cả các phần tử khỏi tập hợp.
 3. **copy()**: Trả về một bản sao nông (shallow copy) của tập hợp.
 4. **difference()**: Trả về hiệu của hai hoặc nhiều tập hợp khác nhau dưới dạng một tập hợp mới.
 5. **difference_update()**: Xóa tất cả các phần tử của một tập hợp khác ra khỏi tập hợp này.
 6. **discard()**: Xóa một phần tử khỏi tập hợp nếu nó là thành viên của tập hợp.
 7. **intersection()**: Trả về giao của hai tập hợp dưới dạng một tập hợp mới.
 8. **intersection_update()**: Cập nhật một tập hợp với giao của chính nó và một tập hợp khác.
 9. **isdisjoint()**: Trả về True nếu hai tập hợp có giao nhau rỗng.
 10. **issubset()**: Trả về True nếu một tập hợp khác chứa tập hợp này.
 11. **issuperset()**: Trả về True nếu tập hợp này chứa một tập hợp khác.
 12. **pop()**: Xóa và trả về một phần tử tùy ý từ tập hợp.
 13. **remove()**: Xóa một phần tử khỏi tập hợp; phần tử đó phải là thành viên của tập hợp.
 14. **symmetric_difference()**: Trả về hiệu đối xứng của hai tập hợp dưới dạng một tập hợp mới.
 15. **symmetric_difference_update()**: Cập nhật một tập hợp với hiệu đối xứng của chính nó và một tập hợp khác.
 16. **union()**: Trả về hợp của các tập hợp dưới dạng một tập hợp mới.
 17. **update()**: Cập nhật một tập hợp với hợp của chính nó và các tập hợp khác.