**1.Đặc điểm:**

* Không chứa các phần tử trùng lặp:Mỗi phần tử trong Set phải là phải là duy nhất.Nếu cố gắng thêm vào một phần tử đã tồn tại thì phần tử ấy sẽ không được thêm vào.
* Không có thứ tự:Set không đảm bảo thứ tự của các phần tử.Tuy nhiên,một số class triển khai Set có thể duy trì thứ tự như(LinkedHashSet).
* Không đồng bộ:Set mặc định không đồng bộ,nghĩa là không an toàn cho đa luồng(multithreading)mà không có xử lý đồng bộ hóa

**2.Các class triển khai từ Set Interface:**

**a.HashSet**

**Đặc điểm:**

* Dựa trên cấu trúc dữ liệu hash table
* Không đảm bảo thứ tự của các phần tử
* Cho phép một phần tử null

**Trường hợp sử dụng:**

* Sử dụng HashSet khi cần một tập hợp không trùng lặp và không cần duy trì thứ tự của các phần tử

**b.LinkedHashSet**

**Đặc điểm:**

* Giống như HashSet,nhưng duy trì thứ tự của các phần tử theo thứ tự mà chúng đã được thêm vào
* Cũng cho phép 1 phần tử null

**Trường hợp sử dụng:**

* Sử dụng LinkedHashSet khi cần một tập hợp không trùng lặp nhưng cần duy trì thứ tự các phần tử

**c.TreeSet**

**Đặc điểm:**

* Dựa trên cấu trúc cây nhị phân tìm kiếm
* Duy trì thứ tự của các phần tử theo thứ tự tự nhiên của chúng
* Không cho phép phần tử null

**Trường hợp sử dụng:**

* Sử dụng treeSet khi cần một tập hợp không trùng lặp và duy trì thứ tự sắp xếp các phần tử

**Một vài phương thức cơ bản:**

* add(E e): Thêm một phần tử vào tập hợp.
* remove(Object o): Xóa phần tử đầu tiên tìm thấy trong tập hợp.
* contains(Object o): Kiểm tra xem tập hợp có chứa phần tử chỉ định hay không.
* size(): Trả về số lượng phần tử trong tập hợp.
* isEmpty(): Kiểm tra xem tập hợp có rỗng hay không.
* clear(): Xóa tất cả các phần tử trong tập hợp.
* iterator(): Trả về một Iterator để duyệt qua tập hợp.