

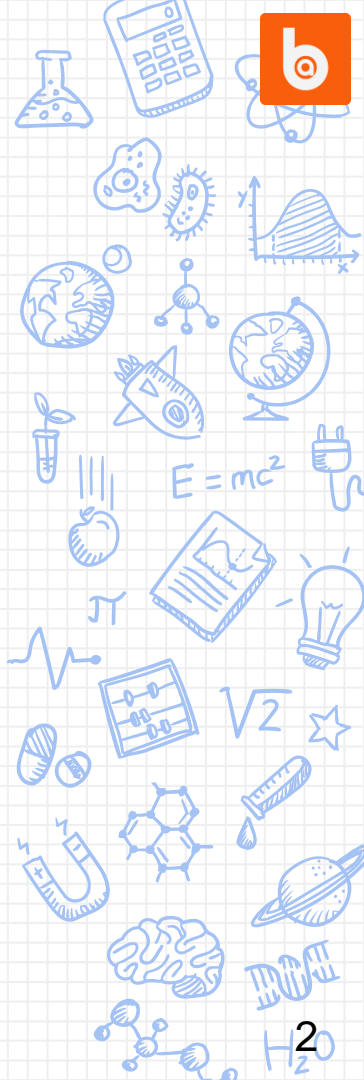
Bài 2.2: Thêm node vào danh sách liên kết đơn

<https://braniumacademy.net>



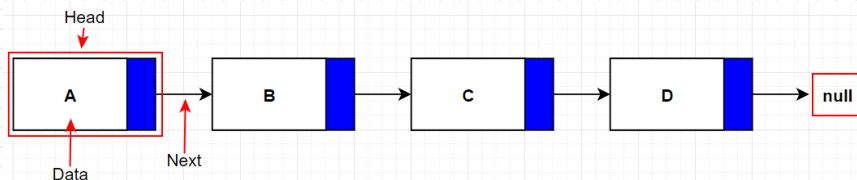
Nội dung bài học

- ☐ Tạo danh sách liên kết đơn
- ☐ Thêm vào đầu danh sách liên kết
- ☐ Thêm vào cuối danh sách liên kết
- ☐ Thêm vào sau node có giá trị x
- ☐ Duyệt danh sách liên kết đơn



Tạo danh sách liên kết

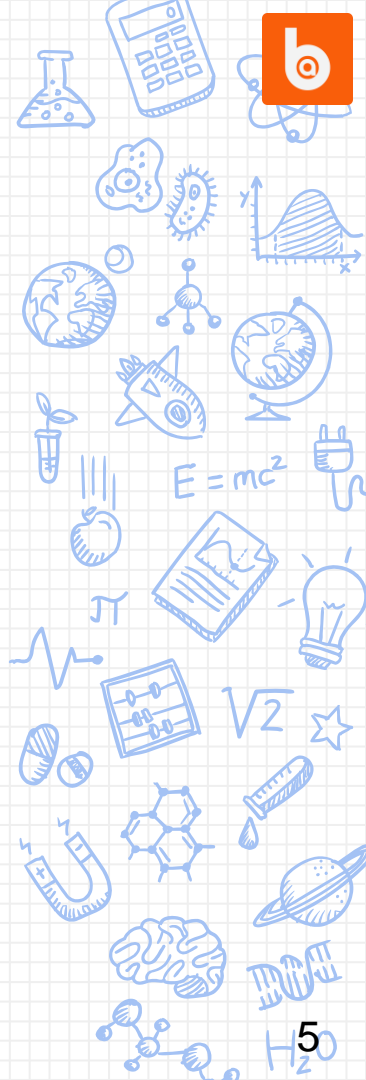
- ❑ Tạo node: inner class tên là Node chứa thuộc tính data, một node next
- ❑ Tạo linked list: outer class đặt tên LinkedList chứa một node head và một node tail



```
public class LinkedList<T> {  
    private Node<T> head;   
    private Node<T> tail;   
  
    static class Node<T> {  
        private T data;   
        private Node<T> next;   
  
        public Node(T data) {  
            this.data = data;  
            this.next = null;  
        }  
    }  
}
```

❑ Giả sử ta muốn chèn node p vào đầu danh sách liên kết.
Các bước thực hiện:

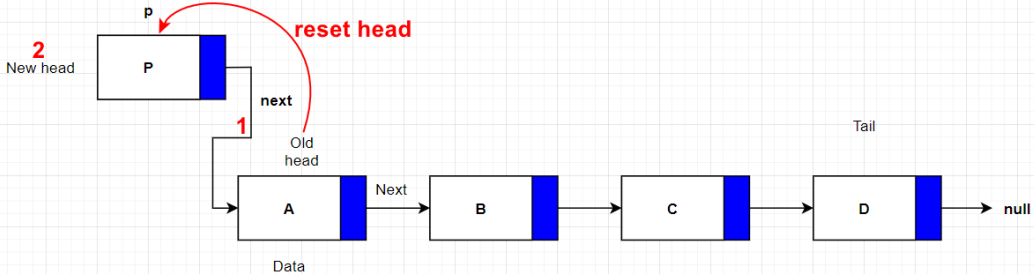
- ✓ Kiểm tra xem node head có null hay không, nếu null ta gán luôn head và tail là p
- ✓ Nếu head khác null:
B1: gán head cho p.next: `p.next = head;`
B2: gán lại head bằng p: `head = p;`



Thêm vào đầu danh sách

□ Triển khai:

```
public void insertHead(T data) {
    Node<T> p = new Node<>(data);
    if (head == null) {
        head = tail = p;
    } else {
        p.next = head;
        head = p;
    }
}
```



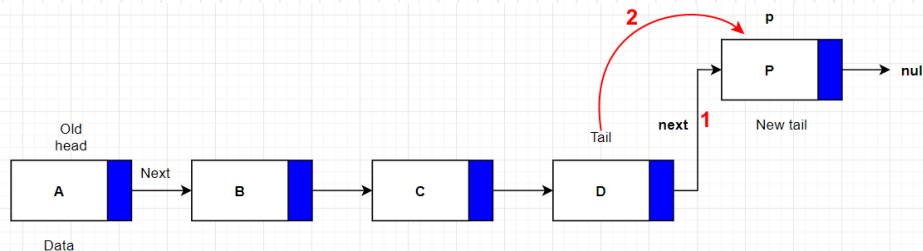
❑ Giả sử ta muốn thêm node p vào cuối danh sách liên kết hiện tại. Các bước thực hiện:

- ✓ Kiểm tra xem head có null hay không. Nếu null ta gán p cho head và tail
- ✓ Nếu head không null, ta thực hiện:
B1: gán p cho tail.next: `tail.next = p;`
B2: gán lại tail: `tail = p;`

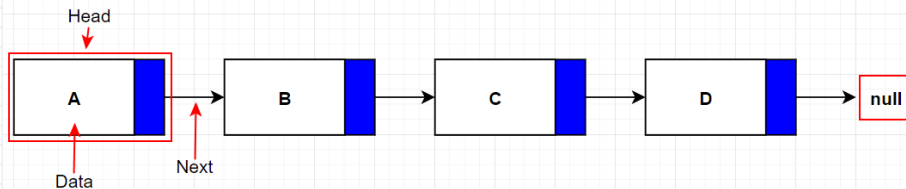
Thêm vào cuối

❑ Triển khai code:

```
public void insertTail(T data) {  
    Node<T> p = new Node<>(data);  
    if (head == null) {  
        head = p;  
        tail = head;  
    } else {  
        tail.next = p;  
        tail = p;  
    }  
}
```



Thêm vào sau node có giá trị x



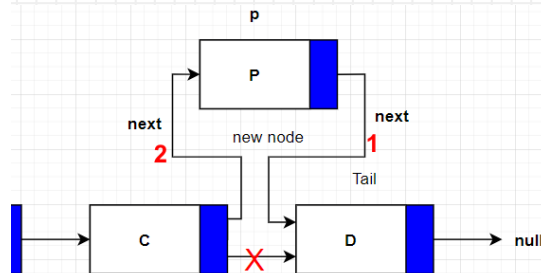
❑ Giả sử ta muốn thêm node p sau node có giá trị C:

- ✓ Tìm node chứa giá trị $X == C$, gọi là nodeX
- ✓ Nếu tìm thấy:
 - B1: gán nodeX.next cho p.next: $p.next = nodeX.next$;
 - B2: gán p cho nodeX.next: $nodeX.next = p$;
- ✓ Nếu không tìm thấy ta thông báo ra màn hình và kết thúc

Thêm vào sau node có giá trị x

❑ Triển khai code:

```
public void insertAfterX(T data, T x) {  
    Node<T> p = new Node<>(data);  
    Node<T> nodeX = head;  
    while (nodeX != null) {  
        if (nodeX.data == x) {  
            break;  
        }  
        nodeX = nodeX.next;  
    }  
    if (nodeX != null) {  
        p.next = nodeX.next;  
        nodeX.next = p;  
    } else {  
        System.out.println("Không tìm thấy node mục tiêu.");  
    }  
}
```



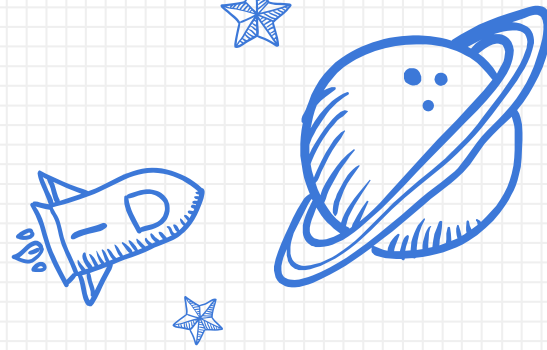
Duyệt danh sách liên kết

❑ Các bước thực hiện:

- ✓ Khai báo node p và khởi tạo bằng head: $p = \text{head}$;
- ✓ Lặp chừng nào p còn khác null:
B1: xuất ra giá trị của p: xuất $p.data$;
B2: cập nhật p: $p = p.next$;

```
public void showList() {  
    for (var node : LinkedList<...>.Node<T> = head; node != null; node = node.next) {  
        System.out.print(node.data + "->");  
    }  
}
```





Tiếp theo

Chèn node vào danh sách liên kết
đôi