**1. Object**

- **data** + **behavior**

Scanner:

* Dữ liệu: luồng ký tự đầu vào (chia thành các token)
* Hành vi: nextInt(), nextLine() ….

String

* Dữ liệu: mảng ký tự cố định
* Hành vi: toUpperCase(), length(), split()….
* Vì sao lớp String không có phương thức nào kiểu *set…. ?*

Như vậy, hành vi được thực hiện dựa trên dữ liệu của đối tượng.

Góc nhìn khác: **hành vi** được hiểu là hành động được **yêu cầu** từ bên ngoài. Và lớp phải có trách nhiệm khai báo, xử lý để **đáp ứng** yêu cầu đó.

**2. DẤU BẰNG**

* Nếu là kiểu dữ liệu cơ bản, dấu bằng là phép gán. Bản chất là copy giá trị và đặt vào vùng nhớ đã được cấp phát cho biến.
* Với khai báo object, dấu bằng được gọi là gán tham chiếu (reference).

Biến object (vế trái) chỉ là một slot trong bộ nhớ (lưu trữ địa chỉ).

Dấu bằng sẽ giúp nó tham chiếu đến (trỏ đến) vùng nhớ chứa data của đối tượng.

* Giá trị **null:** khi object chưa được gán bằng (=) hoặc được gán bằng nhưng giá trị null.

**3. DẤU CHẤM**

* **NullPointerException:** Ngoại lệ rất phổ biến khi code Java
* Bản chất ngoại lệ này xảy ra khi bạn sử dụng dấu chấm mà trước đó chưa có dấu bằng.
* Dấu chấm được gọi là “dereferencing”
* Truy nhập vào bên trong vùng nhớ được tham chiếu đến.

(đi vào trong vùng nhớ đối tượng để yêu cầu thực hiện hành vi)