**1.Kiểu dữ liệu nguyên thủy:**

-Là kiểu dữ liệu cơ bản đc định nghĩa trong Java, kiểu dữ liệu nguyên thủy thường nhanh hơn trong việc xử lý và tiêu tốn ít bộ nhớ hơn so với các kiểu dữ liệu đối tượng,kiểu dữ liệu này không nhận giá trị null,có giá trị khởi tạo.có 8 kiểu như sau:

+ int: Số nguyên 32 bit.

+ byte: Số nguyên 8 bit.

+ short: Số nguyên 16 bit.

+ long: Số nguyên 64 bit.

+ float: Số thực 32 bit.

+ double: Số thực 64 bit.

+ char: Ký tự 16 bit (Unicode).

+ boolean: Giá trị đúng (true) hoặc sai (false).

-Kiểu dữ liệu nguyên thủy đc lưu trữ trực tiếp trên stack,stack hay ngăn xếp là một vùng bộ nhớ mà Java sử dụng để lưu trữ các biến cục bộ và các thông tin về hàm (method) đang thực thi.

**2.Kiểu dữ liệu đối tượng(Object Types):**

-Kiểu dữ liệu đối tượng là các kiểu dữ liệu được định nghĩa từ các lớp (classes). Chúng có thể có trạng thái (thuộc tính) và hành vi (phương thức),giá trị mặc định là null và có thể null,chúng có nhiều kiểu như:

+String,Interger,Long,…

+Interface

+Mảng,Collection

+Một đối tượng với các field đc định nghĩa

-Kiểu dữ liệu Object đc lưu trữ ở Heap khi khởi tạo từ khóa new,heap là một vùng bộ nhớ lớn hơn và có thể lưu trữ các đối tượng. Khi một đối tượng được tạo ra, không chỉ bộ nhớ cho đối tượng mà cả các thuộc tính của nó cũng được cấp phát trên Heap, biến tham chiếu (reference variable) sẽ được lưu trữ trên Stack, và nó sẽ chứa địa chỉ của đối tượng trong Heap

**3.Chuyển đổi kiểu dữ liệu nguyên thủy và đối tượng :**

**a. Từ kiểu nguyên thủy sang kiểu đối tượng (Boxing)**

Trong Java, kiểu dữ liệu nguyên thủy có thể được chuyển đổi thành kiểu đối tượng thông qua quá trình gọi là **boxing**. Điều này có nghĩa là bạn có thể lấy một giá trị nguyên thủy và "đóng gói" nó trong một đối tượng tương ứng.

#### b. Từ kiểu đối tượng sang kiểu nguyên thủy (Unboxing)

Quá trình ngược lại, gọi là **unboxing**, cho phép bạn lấy giá trị nguyên thủy từ một đối tượng.

**4.So sánh kiểu dữ liệu nguyên thủy và đối tượng :**

**a.So sánh kiểu dữ liệu nguyên thủy:**

-Có thể sử dụng các toán tử so sánh như ==,<,>,<=,>=,!= để so sánh các kiểu dữ liệu nguyên thủy như int, byte, short, long, float, double

-Có thể so sánh kiểu dữ liệu Char với các kiểu số học như int,long,… vì trong Java,Char có thể chuyển đổi thành Unicode

-Với Bollean thì chỉ có trả về true/false nên k so sánh đc với các kiểu dữ liệu khác

**b.So sánh kiểu dữ liệu object:**

**-So sánh địa chỉ**: Toán tử == sẽ so sánh địa chỉ của hai đối tượng trong bộ nhớ. Nếu hai biến tham chiếu đến cùng một đối tượng (có cùng địa chỉ), thì kết quả sẽ là true, ngược lại là false(vd: 2 instance của String là str1,str2 đều khởi tạo new String(“helloWorld”) thì sẽ trả về true)

**-So sánh giá trị**: Để so sánh nội dung của hai đối tượng, bạn nên sử dụng phương thức equals(). Phương thức này được định nghĩa trong lớp Object và có thể được ghi đè trong các lớp khác (như String, Integer, v.v.) để so sánh giá trị nội dung.

**c.So sánh primitive với object:**

Khi so sánh đối tượng với kiểu dữ liệu nguyên thủy (ví dụ, int, char), bạn có thể sử dụng phương thức intValue() (hoặc tương tự cho các kiểu khác) để chuyển đổi kiểu đối tượng về kiểu nguyên thủy.