Tìm hiểu về các phép so sánh trong java

Contents

[Khái quát 2](#_Toc71561668)

[DataType primative data type nguyên mẫu. 2](#_Toc71561669)

[Data Type Reference 4](#_Toc71561670)

[DataType Reference đã xây dựng trong java. Integer, String, Float, Double,.... 4](#_Toc71561671)

[DataType Reference tự định nghĩa 6](#_Toc71561672)

# Khái quát

Trong java có 3 kiểu so sánh đặc trưng như sau:

+ Sử dụng toán tử == : return Boolean

Primitive thì so sánh giá trị thực,

Reference đã xây dựng sẵn Sử dụng ô nhớ địa chỉ để xét với nhau.

Reference tự định nghĩa So sánh bộ nhớ địa chỉ.

+ Sử dụng hàm equals() : return Boolean

Primitive so sánh giá trị thực,

Reference đã xây dựng sẵn sử dụng giá trị trực tiếp

Reference tự định nghĩa so sánh ô nhớ nếu ko override.

+ Sử dụng hàm compareTo(): dùng cho sort trả về 1 giá trị số.

Primitive so sánh giá trị thực,

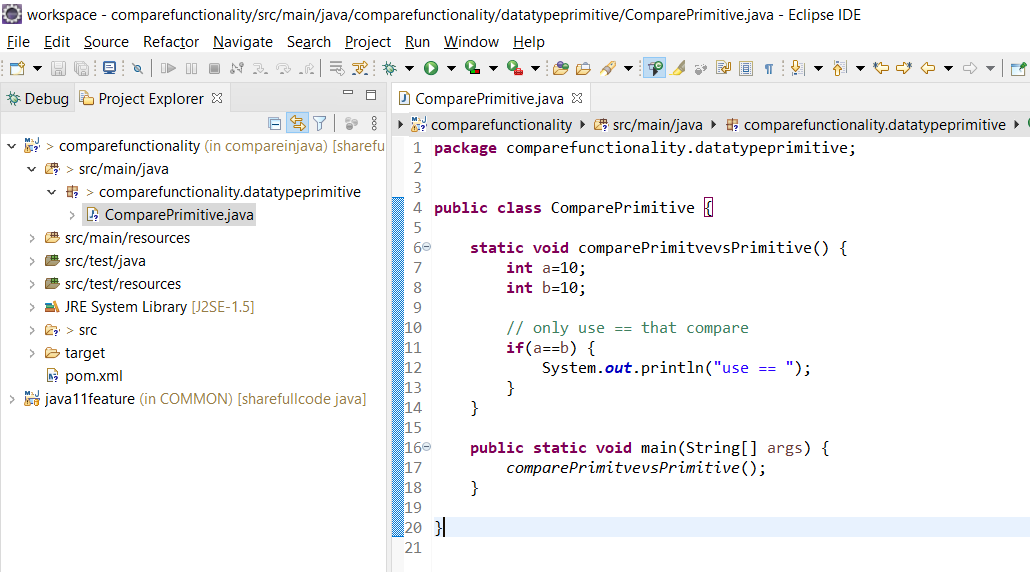
Reference đã xây dựng sẵn sử dụng giá trị trực tiếp,

Reference tự định nghĩa ko có giá trị phải implement Comparable.

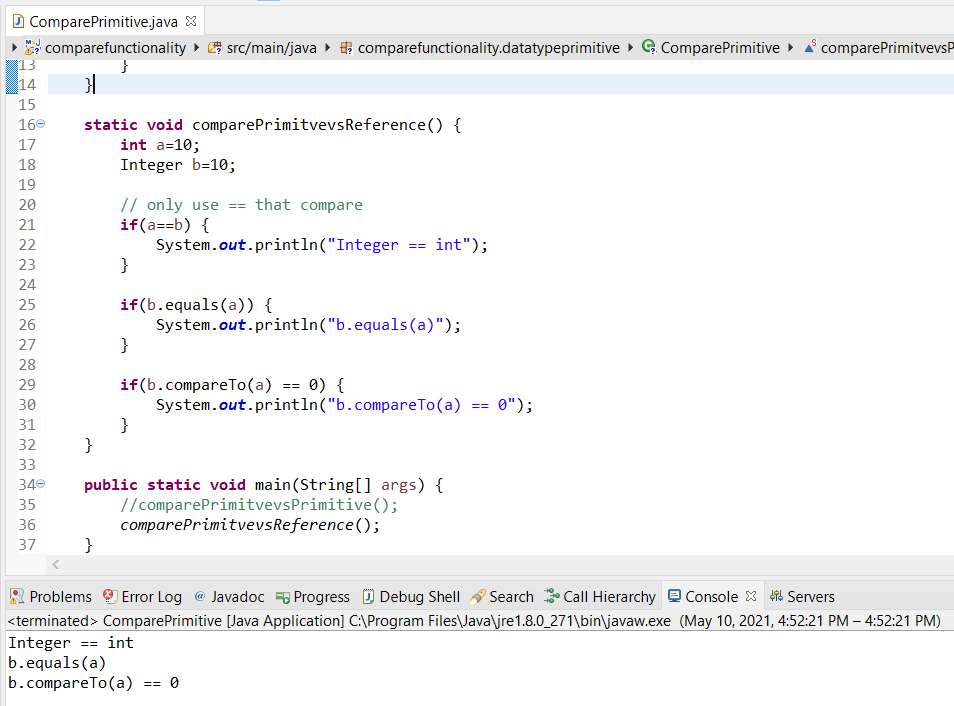
# DataType primative data type nguyên mẫu.

Sử dụng loại nào cũng được.

Primitive (int) vs Primitive (int) Chủ yếu là sử dụng == để so sánh các primitive với nhau.



Primitive (int) vs Reference (Integer) so sánh thì có thể sử dụng hết 3 phương thức đều đúng.



# Data Type Reference

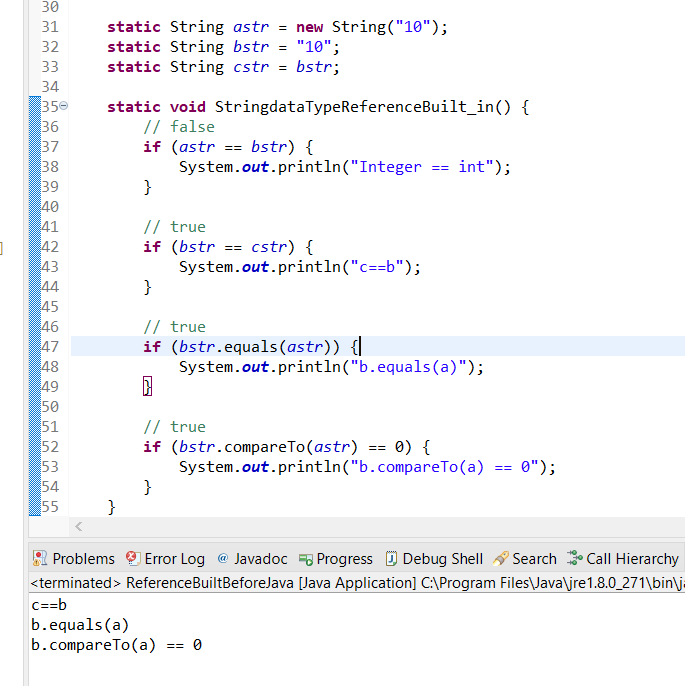
## DataType Reference đã xây dựng trong java. Integer, String, Float, Double,....

Kết luận đó với loại này nên sử dụng equals() hoặc compareTo() để so sánh vì nó so sánh giá trị thật sự mà object đang nắm giữ.

Integer Trong java : == so sánh địa chỉ ô nhớ, equals và compareTo sử dụng Giá trị thật để so sánh.



String trong java cũng giống như interger



## DataType Reference tự định nghĩa

Object tự định nghĩa

== chỉ so sánh địa chị ô nhớ

Equals(): nếu không override thì nó sẽ so sánh ô nhớ.

compareTo: không được định nghĩa sẵn class phải implements Compareable

