Predicate <T> is the type T - > Boolean

Predicate<T> interface is a functional interface that represents a single argument function that returns a boolean value. It is commonly used to evaluate whether a certain condition holds true for a given input.

Predicate is often used for filtering collections, evaluating conditions, or as part of functional programming constructs in Java

T indicates that the Predicate can be used with any type.

Function <T,R> is the type T - > R

The type parameters T and R indicate that the Function can operate on any input type T and return any output type R.

Function is used for transforming data, applying operations to collections, and in functional programming constructs.

Suppliers <T> is the type () - > T

Supplier<T> interface is a functional interface that represents a supplier of results. It is part of the java.util.function package and is used to generate or supply instances of a particular type T.

Unlike Function<T, R>, Supplier does not take any parameters.

Consumer <T> is the type () - > T

Consumer<T> interface is a functional interface that represents an operation that accepts a single input argument and returns no result. It is part of the