***Name:*** Mounika Ponnapalli

***Date:***12-09-2023

**What is the difference between Interface and Functional Interface?**

**i)** **Normal Interface :**

A normal interface can contain one or more abstract methods.

It can also have any number of static and default methods, and it can declare variables that are public static final (constants) by default.

It can include public methods from the Object class because all Java classes implicitly extend Object.

**ii) Functional Interface:**

A functional interface is a type of interface that has exactly one abstract method (SAM - Single Abstract Method).

Similar to a normal interface, it can also have any number of static and default methods, and it can include public methods from the Object class.

Functional interfaces are designed to be used with lambda expressions and method references.

**Normal Interface (Chocolate Interface):**

Imagine it's like asking every chocolate brand (like Cadbury, Nestle) to answer two questions: How does your chocolate taste, and how much does it cost?

We don't force them to answer in a specific way; they can tell us whatever they think is right.

This interface sets the rules: You must tell us these two things, but it doesn't tell them exactly how to do it.

**Functional Interface (ChocolateComparator):**

Think of it as a taste test competition for chocolates.

Different judges are there, and each judge can use their own criteria to decide which chocolate is the best - sweetness, bitterness, or anything else they like.

So, it's like having a panel of judges, and each judge has their own rules for picking the best chocolate.

In this example, the Chocolate Interface ensures that chocolate brands provide specific information, while the ChocolateComparator allows users to find the best chocolate based on their individual tastes, just like in a real-world taste test competition.