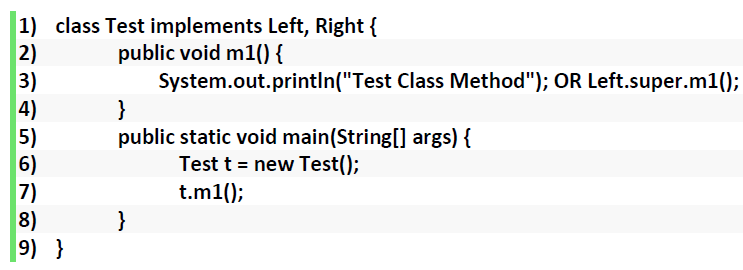
* **Functional Interface** – only one abstract method - @FunctionalInterface

Functional Interface can extend another Functional Interface only if child does not have abstract method

* **Marker Interface** – with no method inside it, blank interface, like, **Cloneable**, **Serializable, Remote Interface.**
* **Default Method / Static Method** – included in Java8, Static method only can be called by **InterfaceName.MethodName()** … by implementation object name it is not allowed, throw a compile time error. Default method can override… if **ambiguity** occurs then always need to override and inside call super one by following.



**Function-** apply() – applyAsDouble()

**Predicate**- test()

**Supplier** – get() – getAsInt () -can’t return 2 value in java, hence BiSupplier not available.

**Consumer**- accept()

Java Optional Class

Java introduced a new class Optional in **jdk8**. It is a public final class and used to deal with **NullPointerException** in Java application. You must import **java.util** package to use this class.

**Stream() - 1. Intermediate Operations and 2. Terminal Operations**

1. map , filter , sorted
2. Count, forEach , collect, reduce

**Method Reference:** Method reference is also known by “::”. 1. If method is static then **Class\_Name::Method\_Name** 2. if method is non-static then **Object\_Name::Method\_Name**