**Java-8 – Functional Interface Annotation with respect to Inheritance:**

* **Case\_01:**

If an interface extends Functional Interface annotation and child interface doesn’t contain any abstract method, then child interface is always functional interface.

Ex:

@FunctionalInterface

interface P{

public void m1();

}

@FunctionalInterface

interface C extends P{

}

* **Case\_02:**

In the child interface, we can define exactly same parent interface abstract method.

Ex:

@FunctionalInterface

interface P{

public void m1();

}

@FunctionalInterface

Interface C extends P{

public void m1();

}

* **Case\_03:**

In the child interface we can’t define any new abstract methods otherwise we will get CE.

Ex:

@FunctionalInterface

interface P{

public void m1();

}

@FunctionalInterface

interface C extends P{

public void m2();

}

CE: Unexpected @FunctionalInterface annotation

Multiple non-overriding abstract methods found in interface C

* **Case\_04:**

@FunctionalInterface

interface P{

public void m1();

}

interface C extends P{

public void m2();

}

This case is valid as we didn’t declare the child class as @FunctionalInterface