**Java 8 Features**

1. **Default methods(defining a method in interface) and Static methods in interface**

**package** com.feature.first;

**public** **interface** Test {

**void** display(); // by default this method is public abstract

**default** **void** show() { // using default keyword we can define a method in interface, from java 1.8

System.***out***.println("new feature in java 8.....");

}

**default** **void** printMessage() {

System.***out***.println("testing new feature....");

}

**static** **void** success() { // from java 1.8 we can have static methods in interface

System.***out***.println("Transaction success.....");

}

}

**package** com.feature.first;

**public** **class** TestImpl **implements** Test{

// public void show() { // if we want we can override the default methods here

// System.out.println("overriding the message in interface Test ....");

// }

@Override

**public** **void** display() {

System.***out***.println("in display method ....");

}

**public** **static** **void** main(String[] args) {

TestImpl obj = **new** TestImpl();

obj.show();

Test.*success*(); // since success is a static method, we can directly call this method with interface name

}

}

**2. ) for-Each method**

**package** com.feature.second;

**import** java.util.Arrays;

**import** java.util.List;

**public** **class** SecondFeature {

**public** **static** **void** main(String[] args) {

List<Integer> list = Arrays.*asList*(1,2,3,4,5);

// for(int i=0;i<list.size();i++) { // iterate the list using for loop

// System.out.println("Printing the elements in the list : "+list.get(i));

// }

// for(int j : list) { // iterate the list using for each loop

// System.out.println("Printing the elements in the list : "+j);

// }

list.forEach(k -> System.***out***.println("Printing the elements in the list : "+k)); // iterate the list using forEach method, this forEach method is available since java 1.8

}

}