**Java-8 – Static methods inside interface**

* **Static methods inside interface:**

From 1.8 version onwards, we can define static method inside interface.

We can use this method as a utility.

interface Interf{

public static void m1(){

System.out.println(“interface static method”);

}

}

Note:

Interface static methods by default not available to the implementation class.

We have to call the static methods directly with the interface name, not using the reference name. Otherwise we will get compile time error.

class Test implements Interf{

public static void main(String[] args){

m1(); // cannot find symbol

Test t = new Test();

t.m1(); // Invalid, cannot find symbol

Test.m1(); // Invalid

Interf.m1(); // Valid

}

}

* **Interface static methods with respect to overriding:**

Interface static methods by default not available to the static class, hence override future is not applicable for interface static methods.

interface Interf{

public static void m1(){

}

}

class Test implements Interf{

public static void m1(){

}

}

Note: It’s valid, but it is not overriding.

interface Interf{

public static void m1(){

}

}

class Test implements Interf{

public void m1(){

}

}

Note: Valid, but not overriding.

interface Interf{

public static void m1(){

}

}

class Test implements Interf{

private static void m1(){

}

}

Note:Valid, but not overriding.

* **main() inside inteface:**

We can declare a static method inside an interface, hence we can declare main method also and we can run the main method from command prompt.

interface Interf{

public static void main(String[] args){

System.out.println(“Interface main method”);

}

}