TestDemo1.java

```
package com.java8features;
public class TestDemo1 {
    public static void main(String[] args) {
        TestInterf1 t1 = new TestInterf1Impl1();
        t1.method1();
        t1.method2();
        t1.method3();
        TestInterf1.metho5();

        TestInterf1 t2 = new TestInterf1Impl2();
        t2.method1();
        t2.method2();
        t2.method3();
        TestInterf1 t3 = new TestInterf1Impl3();
        t3.method1();
        t3.method2();
        t3.method3();
        TestInterf1.metho5();

        TestInterf1Impl3.method5();


    }
}
```

TestDemo2.java

```java
package com.java8features;
@FunctionalInterface
interface Testinterf2 {
   void hello();
   default void welcome() {
      System.out.println("Welcome interf1 !!");
   }
}
public class TestDemo2 {
   public static void main(String[] args) {
      Testinterf2 t2 = () -> {
         System.out.println("Hello guys good morning !!");
         System.out.println("Hello guys good morning !!");
         System.out.println("Hello guys good morning !!");
      };
      t2.hello();
      t2.welcome();
   }
}
```

TestDemo3.java

```
package com.java8features;
interface TestInterf3 {
   int hello(int a, int b);
}
public class TestDemo3 {
   public static void main(String[] args) {
      TestInterf3 t3 = (s, s1) -> {
         int s3 = s + s1;
         return s3;
      };
      int total = t3.hello(100, 200);
      System.out.println(total);
   }
}
```

TestDemo4.java

```
package com.java8features;
```

```
interface TestInterf4 {
   int squareit(int a);
}
public class TestDemo4 {
   public static void main(String[] args) {
      TestInterf4 t4 = (a) -> {
         return a * a;
      };
      int sqrt = t4.squareit(10);
      System.out.println(sqrt);
   }
}
```

TestDemo5.java

```
package com.java8features;
public class TestDemo5 {
   public static void main(String[] args) {
      System.out.println("Hello main method
!!");
      Runnable r = () -> {
         for (int i = 100; i < 200; i++) {
            System.out.println("run :" + i);
         }
      };
```

```
        Thread t = new Thread(r);
        t.start();
        for (int i = 0; i <= 100; i++) {
            System.out.println("main " + i);
        }
    }
}
```

TestDemo6.java

```
package com.java8features;
@FunctionalInterface
interface TestInterf5 {
    public void method1();
}
@FunctionalInterface
interface TestInterf6 extends TestInterf5 {
}
public class TestDemo6 {
    public static void main(String[] args) {
        TestInterf6 t6 = () ->
System.out.println("Hello method1");
        t6.method1();
    }
}
```

TestInterf1.java

```
package com.java8features;
public interface TestInterf1 {
   public static void main(String[] args) {
      System.out.println("Hello main !!");
   }
   // upto 1.7
   // by default all interface variables are
static and final
   static final int x = 100;
   // by default all methods in interfaces
public and abstract
   public abstract void method1();
   public abstract void method2();
   // 1.8 version onwards
   // default methods in interfaces
   // static methods
   // Note : For default methods private
access modifier is not possible

   public default void method3() {
      System.out.println("interf method3 !!");
   }
   public default void method4() {
        System.out.println("interf method4 !!");
        method6();
    }
    public static void metho5() {
        System.out.println("interf method5 !!");
//      method6();
    }
    // 1.9 onwards onwards
    // private methods
    private void method6() {
        System.out.println("Hello private 
method6 !!");
    }
}
```

TestInterf1Impl1.java

```
package com.java8features;
public class TestInterf1Impl1 implements 
TestInterf1 {

    @Override
    public void method1() {
        System.out.println("Hello method1 !!");
    }
    @Override
    public void method2() {
        System.out.println("Hello method2 !!");
    }
    @Override
    public void method3() {
        System.out.println("Hello method3 !!");
    }
    public static void method5() {
        System.out.println("Hello method5 ");
    }
}
```

TestInterf1Impl2.java

```
package com.java8features;
public class TestInterf1Impl2 implements
TestInterf1 {
    @Override
    public void method1() {
        System.out.println("Hiiii method1 !!");
    }

    @Override
    public void method2() {
        System.out.println("Hiiii method2 !!");
    }
}
```

TestInterf1Impl3.java

```
package com.java8features;
public class TestInterf1Impl3 implements
TestInterf1 {
    @Override
    public void method1() {
        System.out.println("welcome method1
!!");
    }
    @Override
    public void method2() {
        System.out.println("welcome method2
!!");
    }
    public static void method5(){
        System.out.println("hello hiding");
    }
}
```

TestDemo7.java

```java
package com.java8features;
import java.util.ArrayList;
import java.util.Collections;
import java.util.List;
public class TestDemo7 {
    public static void main(String[] args) {
        List<Integer> al = new ArrayList<>();
        al.add(6);
        al.add(5);
        al.add(4);
        al.add(3);
        al.add(10);
        al.add(15);
        al.add(67);
        al.add(18);
        System.out.println(al);
        Collections.sort(al);
        System.out.println(al);
    }
}
```

Employee.java

```java
package com.collectionsframework;
public class Employee {
   private int eid;
   private String ename;
   private double esal;
   @Override
   public String toString() {
      return "Employee [eid=" + eid + ",
ename=" + ename + ", esal=" + esal + "]";
   }
   public Employee(int eid, String ename,
double esal) {
      super();
      this.eid = eid;
      this.ename = ename;
      this.esal = esal;
   }
   public int getEid() {
      return eid;
   }
   public void setEid(int eid) {
      this.eid = eid;
   }
   public String getEname() {
        return ename;
    }
    public void setEname(String ename) {
        this.ename = ename;
    }
    public double getEsal() {
        return esal;
    }
    public void setEsal(double esal) {
        this.esal = esal;
    }
}
```

TestEmployee.java

```
package com.collectionsframework;
package com.collectionsframework;
import java.util.ArrayList;
import java.util.Collections;
import java.util.Comparator;
import java.util.List;
public class TestEmployee {
    public static void main(String[] args) {
        Employee e1 = new Employee(101,
"BabuRam", 60000.00);
```

```
      Employee e2 = new Employee(104,
"Govind", 40000.00);
      Employee e3 = new Employee(102, "Vinay",
80000.00);
      Employee e4 = new Employee(103, "Veera",
160000.00);
      Employee e5 = new Employee(105,
"Avinash", 260000.00);
      List<Employee> empList = new
ArrayList<>();
      empList.add(e1);
      empList.add(e2);
      empList.add(e3);
      empList.add(e4);
      empList.add(e5);
      Comparator<Employee> c = (o1, o2) ->
(o1.getEid() < o2.getEid()) ? -1 :
(o1.getEid() > o2.getEid()) ? 1 : 0;
      Collections.sort(empList, c);
      for (Employee emp : empList) {
         System.out.println(emp);
      }
   }
}
```