冒泡排序：

public class Test3 {  
 public static void main(String[] args) {  
 int[] arr = {23,45,12,46,56,123,45,32,76,79,56};  
 int temp;  
 for(int i=0;i<arr.length-1;i++){  
 for(int j=0;j<arr.length-i-1;j++){  
 if(arr[j]>arr[j+1]){  
 temp = arr[j];  
 arr[j] = arr[j+1];  
 arr[j+1] = temp;  
 }  
 }  
 }  
 for(int i : arr){  
 System.*out*.print(i + " ");  
 }  
 }  
}

饿汉模式：

package compare;  
  
public class Hungry {  
 private static Hungry *demo* = new Hungry();  
 private Hungry(){  
  
 }  
 public static Hungry getDemo() {  
 return *demo*;  
 }  
  
}

懒汉模式：

package compare;  
  
public class Lazy {  
 private static volatile Lazy *demo*;  
 private Lazy(){  
  
 }  
 public static Lazy getDemo(){  
 if(*demo*==null){  
 synchronized (Lazy.class){  
 if(*demo*==null){  
 *demo* = new Lazy();  
 }  
 }  
 }  
 return *demo*;  
 }  
}

优化之静态内部类：

package compare;  
  
public class inner {  
 private inner(){  
  
 }  
 private static class Demo{  
 private static inner *demo* = new inner();  
 }  
 public static inner getDemo(){  
 return Demo.*demo*;  
 }  
}