选择排序：假设数组长度为lenght，选择排序的外循环也是lenght-1，内循环则为使用第一个数与其他的数比较，得出一个最小的数，将最小数与数组的第一个元素交换；第二次外循环，则用数组的第二个数与第三到底lenght个数比较，找到其中的最小值，并将此最小值与数组的第二个数交换，一直下去，直到数组末尾。

**package** facehandjava.sort;  
  
**public class** SelectionSort {  
 **public static void** main(String[] args) {  
 **int**[] arrays = {10, 5, 36, 78,56,2, 5, 8, 9, 9};  
  
 **int** min = 0;  
 **int** max = arrays.**length**-1;  
 System.***out***.print(**"原来的："**);  
 **for**(**int** i =0;i<=max;i++) {  
 System.***out***.print(arrays[i]+**","**);  
 }  
 System.***out***.println();  
 *SelectionSort*(arrays,max);  
 System.***out***.print(**"排序后："**);  
 **for**(**int** i =0;i<=max;i++) {  
 System.***out***.print(arrays[i]+**","**);  
 }  
  
 }  
  
 **public static void** SelectionSort(**int**[] arrays,**int** max) {  
 **for** (**int** i = 0; i <= max; i++) {  
 **for** (**int** j = i+1; j <= max; j++) {  
 **if** (arrays[i] > arrays[j]) {  
 *swap*(arrays, i, j);  
 }  
 }  
 }  
 }  
  
  
 **public static void** swap(**int**[] arrays,**int** a, **int** b) {  
 **int** temp = arrays[a];  
 arrays[a] = arrays[b];  
 arrays[b] = temp;  
 }  
}