**Superset ID:**  6125132  
**Name:** Podugu Vinay Kumar

**Week-1:**

**Exercise 3: Sorting Customer Orders**

**Input Code:  
  
Order.java:**package ordersort;

public class Order {

private String orderId;

private String customerName;

private double totalPrice;

public Order(String orderId, String customerName, double totalPrice) {

this.orderId = orderId;

this.customerName = customerName;

this.totalPrice = totalPrice;

}

public double getTotalPrice() {

return totalPrice;

}

@Override

public String toString() {

return "Order ID: " + orderId + ", Customer: " + customerName + ", Total Price: $" + totalPrice;

}

}  
  
**Sort Algorithms.java :**

package ordersort;

public class SortAlgorithms {

public static void bubbleSort(Order[] orders) {

int n = orders.length;

for (int i = 0; i < n - 1; i++) {

for (int j = 0; j < n - 1 - i; j++) {

if (orders[j].getTotalPrice() > orders[j + 1].getTotalPrice()) {

Order temp = orders[j];

orders[j] = orders[j + 1];

orders[j + 1] = temp;

}

}

}

}

public static void quickSort(Order[] orders, int low, int high) {

if (low < high) {

int pivotIndex = *partition*(orders, low, high);

*quickSort*(orders, low, pivotIndex - 1);

*quickSort*(orders, pivotIndex + 1, high);

}

}

private static int partition(Order[] orders, int low, int high) {

double pivot = orders[high].getTotalPrice();

int i = low - 1;

for (int j = low; j < high; j++) {

if (orders[j].getTotalPrice() <= pivot) {

i++;

Order temp = orders[i];

orders[i] = orders[j];

orders[j] = temp;

}

}

Order temp = orders[i + 1];

orders[i + 1] = orders[high];

orders[high] = temp;

return i + 1;

}

public static void printOrders(Order[] orders) {

for (Order order : orders) {

System.*out*.println(order);

}

}

}  
  
**Main.java :  
  
package** ordersort;

**public** **class** main {

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

Order[] orders = {

**new** Order("01", "Alice", 450.75),

**new** Order("02", "Bob", 199.99),

**new** Order("03", "Charlie", 700.50),

**new** Order("04", "David", 300.00)

};

System.***out***.println("Original Orders:");

SortAlgorithms.*printOrders*(orders);

Order[] bubbleSorted = orders.clone();

SortAlgorithms.*bubbleSort*(bubbleSorted);

System.***out***.println("\n Orders Sorted by Bubble Sort:");

SortAlgorithms.*printOrders*(bubbleSorted);

Order[] quickSorted = orders.clone();

SortAlgorithms.*quickSort*(quickSorted, 0, quickSorted.length - 1);

System.***out***.println("\n Orders Sorted by Quick Sort:");

SortAlgorithms.*printOrders*(quickSorted);

}

}

