18. Create a collection named **“ORDERS”** that contain documents of the following prototype and solve the following queries:

```
{
        cust_id: "abc123",
        ord_date: new Date("Oct 04, 2012"),
        status: 'A',
        price: 50,
        items: [ { sku: "xxx", qty: 25, price: 1 },
                { sku: "yyy", qty: 25, price: 1 } ]
}
```

a. Count all records from `orders`

b. Sum the `price` field from `orders`

c. For each unique cust_id, sum the price field.

d. For each unique cust_id, sum the price field, results sorted by sum.

For each unique cust_id, ord_date grouping, sum the price field

```
test> db.orders.insertOne({cust_id:"sdf566",ord_date:new Date("Nov 11,2021"),status:'A',price:70,items:[{sku:"bbb",qty:45,price:2},{sku:"ccc",qty:50,price:9}]});
```

```
test> db.orders.aggregate([{$group:{_id:null,total:{$sum:1}}}]);
```

```
test> db.orders.aggregate([{$group:{_id:null,total_amount:{$sum:"$price"}}}]);
```

```
test> db.orders.aggregate([{$group:{_id:"$cust_id",total_amount:{$sum:"$price"}}}]);
```

```
test>db.orders.aggregate([{$group:{_id:"$cust_id",total_amount:{$sum:"$price"}}},{$sort:{total_amount:-1}}]);
```

```
test>db.orders.aggregate([{$group:{_id:"$cust_id",total_amount:{$sum:"$price"}}},{$sort:{total_amount:1}}]);
```

```
test>db.orders.aggregate([{$group:{_id:{cust_id:"$cust_id",ord_date:"$ord_date"},total_amount:{$sum:"$price"}}}]);
```