

## ◆ Question 1 — FILTER

For each order, create a field `bulkItems` that keeps only items where `qty ≥ 2`.

### ✓ Answer

```
db.orders.aggregate([ { $project: { orderId: 1, bulkItems: { $filter: { input: "$items", as: "i", cond: { $gte: ["$$i.qty", 2] } } } } }]);
```

## ◆ Question 2 — MAP

For each order, create `skuList` containing only SKUs of items.

### ✓ Answer

```
db.orders.aggregate([ { $project: { orderId: 1, skuList: { $map: { input: "$items", as: "i", in: " $$i.sku" } } } }]);
```

## ◆ Question 3 — REDUCE

For each order, compute `orderTotal = Σ(qty × price)`.

### ✓ Answer

```
db.orders.aggregate([ { $project: { orderId: 1, orderTotal: { $reduce: { input: "$items", initialValue: 0, in: { $add: [ "$$value", { $multiply: [ "$$this.qty", "$$this.price" ] } ] } } } } ]);
```

## ◆ Question 4 — FILTER + MAP

For each order, create `highSkus`:  
SKUs of items where `price ≥ 200`.

### ✓ Answer

```
db.orders.aggregate([ { $project: { orderId: 1, highSkus: { $map: { input: { $filter: { input: "$items", as: "i", cond: { $gte: ["$$i.price", 200] } } }, as: "x", in: " $$x.sku" } } } ]);
```

## ◆ Question 5 — FILTER + REDUCE

For each order, compute  $\text{highValueTotal} = \sum(\text{qty} \times \text{price})$   
ONLY for items where  $\text{price} \geq 200$ .

### ✓ Answer

```
db.orders.aggregate([ { $project: { orderId: 1, highValueTotal: { $reduce: { input: { $filter: { input: "$items", as: "i", cond: { $gte: ["$$i.price", 200] } } }, initialValue: 0, in: { $add: [ "$$value", { $multiply: [ "$$this.qty", "$$this.price" ] } ] } } } } }]);
```