이 MongoDB 쿼리들은 orders 컬렉션에서 데이터를 집계하여 다양한 정보를 추출합니다. 아래는 각 쿼리의 설명입니다:

1. 피자 크기별 주문 수량 집계

```
db.orders.aggregate([
    // Stage 1: 피자 크기로 필터링
    {
        $match: { size: "medium" }
    },

    // Stage 2: 피자 이름별로 그룹화하고 총 수량 계산
    {
        $group: { _id: "$name", totalQuantity: { $sum: "$quantity" } }
    }
}
```

- Stage 1: size 가 "medium" 인 주문만 필터링합니다.
- Stage 2: 필터링된 문서들을 name 으로 그룹화하여 각 피자 이름별로 총 수량(totalQuantity)을 계산합니다.

이 쿼리는 size 가 "medium" 인 피자 주문의 총 수량을 피자 이름별로 계산합니다. 결과는 각 피자 이름과 해당 피자 이름의 "medium" 사이즈 주문 총 수량을 포함합니다.

2. 날짜 범위 내 총 주문 가치와 평균 주문 수량 계산

```
},

// Stage 3: 총 주문 가치를 기준으로 내림차순 정렬
{
     $sort: { totalOrderValue: -1 }
}
```

- Stage 1: date 가 2020-01-30 부터 2022-01-30 사이의 문서만 필터링합니다.
- Stage 2: 필터링된 문서들을 날짜별로 그룹화하고, 각 날짜에 대해 총 주문 가치 (totalOrderValue)와 평균 주문 수량(averageOrderQuantity)을 계산합니다. 총 주문 가치는 price와 quantity를 곱하여 합산합니다.
- Stage 3: 결과를 totalOrderValue 기준으로 내림차순으로 정렬합니다.

이 쿼리는 주어진 날짜 범위 내에서 각 날짜별 총 주문 가치와 평균 주문 수량을 계산하며, 총 주문 가치가 높은 순서대로 정렬합니다.

전체적인 설명

이 두 가지 집계 쿼리는 MongoDB의 Aggregation Pipeline을 사용하여 복잡한 데이터 집계를 수행합니다. 첫 번째 쿼리는 특정 크기(중간)의 피자 주문 수량을 집계하는 데 사용되고, 두 번째 쿼리는 특정 날짜 범위 내에서 각 날짜별 주문의 총 가치를 계산하고 이를 내림차순으로 정렬하는 데 사용됩니다. MongoDB의 Aggregation Pipeline을 사용하면 데이터의 복잡한 분석과 집계를 효율적으로 수행할 수 있습니다.