

```
$match → $project(concat / compute) → $group(distinct) → $project(clean)
```

```
db.orders1.find();
// 1. display the first_name,last_name, with delimiter(space) as Name of the
records
// where last_name having only one letter. records should be distinct
// ex: {"Name":"Tarun GA"}
db.orders1.aggregate([
  // match last name with exactly one character
  { $match: { last_name: { $regex: /^.$/ } } },
  {
    // concatenate first_name+last name
    $project: {
      _id: 0, Name: { $concat: ["$first_name" , " " , "$last_name"] }
    }
  },
  // makes names distinct
  { $group: { _id: "$Name" } },
  { $project: { _id: 0, Name: "$_id" } }
]);
```

```
// 2. display customer name and number of orders ordered by each customer
db.orders1.aggregate([
  {
    $group: {
      _id:"$first_name",
      // if they ask full name
      // _id: { fullName: { $concat: ["$first_name", " ", "$last_name"] } },
      orderCount: { $sum: 1 }
    }
  },
  {
    $project: {
      _id: 0, CustomerName: "$_id", orderCount: 1
    }
  }
]);
```

```
// 3. display top 5 customers based on total money spent on their orders
db.orders1.aggregate([
  {
    $group: {
      _id: "$first_name", totalMoney: { $sum: "$amount" }, // grouping 1st
name is ok,but better full name(by concatenating first_name+last_name)
    }
  },
  {
    $sort: { totalMoney: -1 }
  },
  { $limit: 5 }
]);
```

```
{ $sort: { totalMoney: -1 } },
{ $limit: 5 },
{
  $project: {
    _id: 0,
    CustomerName: "$_id.fullName",
    totalMoney: 1
  }
}

]);

// 4. display the number of unique customers in each gender
db.orders1.aggregate([
  { $group: { _id: "$gender", uniqueCustomers: { $addToSet: "$first_name" } } },
  // here name bcz same person can order 2 or more times
  {
    $project: {
      _id: 0, Gender: "$_id", uniqueCustomersCount: { $size:
"$uniqueCustomers" }
    }
  }
]);
// 5,6 these are sql to mongodb syntax conversion
```