SQL and MongoDB Queries:

1. SELECT shipVia, COUNT(\*) AS total\_shipments FROM orders GROUP BY shipVia ORDER BY total\_shipments DESC LIMIT 1;

db.orders.aggregate([

{

$group: {

\_id: "$shipVia",

total\_shipments: { $sum: 1 }

}

},

{

$sort: {

total\_shipments: -1

}

},

{

$limit: 1

}

]);

1. Left
2. Sql : SELECT lastName FROM employees WHERE MONTH(birthDate) = 11;

MongoDB :

db.employees.find(

{

$expr: {

$eq: [{ $month: "$birthDate" }, 11]

}

},

{

lastName: 1

}

);

1. SQL :

SELECT e.lastName, e.firstName, t.territoryDescription AS Territory FROM employees e JOIN employee\_territories et ON e.employeeID = et.employeeID JOIN territories t ON et.territoryID = t.territoryID ORDER BY t.territoryDescription, e.lastName;

MongoDB :   
 db.employees.aggregate([

{

$lookup: {

from: "employee\_territories",

localField: "employeeID",

foreignField: "employeeID",

as: "territories"

}

},

{

$unwind: "$territories"

},

{

$lookup: {

from: "territories",

localField: "territories.territoryID",

foreignField: "territoryID",

as: "territory"

}

},

{

$unwind: "$territory"

},

{

$project: {

\_id: 0,

lastName: 1,

firstName: 1,

Territory: "$territory.territoryDescription"

}

},

{ $sort: {

territory: 1,

lastName: 1 } } ]);

1. SQL :

SELECT p.productName, SUM(od.quantity \* od.unitPrice) AS TotalSales FROM products p JOIN order\_details od ON p.productID = od.productID GROUP BY p.productName ORDER BY TotalSales DESC LIMIT 1;

MySql :   
db.products.aggregate([

{

$lookup: {

from: "order\_details",

localField: "productID",

foreignField: "productID",

as: "orderDetails"

}

},

{

$unwind: "$orderDetails"

},

{

$group: {

\_id: "$productName",

TotalSales: { $sum: { $multiply: ["$orderDetails.quantity", "$orderDetails.unitPrice"] } }

}

},

{

$sort: {

TotalSales: -1

}

},

{

$limit: 1

}

]);

1. Sql :   
   SELECT p.productName, COALESCE(SUM(od.quantity \* od.unitPrice), 0) AS totalSales FROM products p LEFT JOIN orderDetails od ON p.productID = od.productID GROUP BY p.productName ORDER BY totalSales LIMIT 1;

MongoDB :

db.products.aggregate([

{

$lookup: {

from: "orderDetails",

localField: "productID",

foreignField: "productID",

as: "orderDetails"

}

},

{

$group: {

\_id: "$productName",

totalSales: { $sum: { $multiply: [{ $ifNull: ["$orderDetails.quantity", 0] }, { $ifNull: ["$orderDetails.unitPrice", 0] }] } }

}

},

{

$sort: {

totalSales: 1

}

},

{

$limit: 1

}

]);

1. Sql :

SELECT MONTH(orderDate) AS month, SUM(quantity \* unitPrice) AS totalSales FROM orderDetails JOIN orders ON orderDetails.orderID = orders.orderID GROUP BY month ORDER BY totalSales DESC LIMIT 1;

MongoDB :

db.orderDetails.aggregate([

{

$lookup: {

from: "orders",

localField: "orderID",

foreignField: "orderID",

as: "order"

}

},

{

$unwind: "$order"

},

{

$group: {

\_id: { $month: "$order.orderDate" },

totalSales: { $sum: { $multiply: ["$quantity", "$unitPrice"] } }

}

},

{

$sort: {

totalSales: -1

}

},

{ $limit: 1 } ]);

1. SQL :  
   SELECT e.firstName, e.lastName, SUM(od.quantity \* od.unitPrice) AS totalSales FROM employees e JOIN orders o ON e.employeeID = o.employeeID JOIN orderDetails od ON o.orderID = od.orderID GROUP BY e.firstName, e.lastName ORDER BY totalSales DESC

LIMIT 1;

MongoDB :

db.employees.aggregate([

{

$lookup: {

from: "orders",

localField: "employeeID",

foreignField: "employeeID",

as: "orders"

}

},

{

$unwind: "$orders"

},

{

$lookup: {

from: "orderDetails",

localField: "orders.orderID",

foreignField: "orderID",

as: "orderDetails"

}

},

{

$unwind: "$orderDetails"

},

{

$group: {

\_id: { employeeID: "$employeeID", firstName: "$firstName", lastName: "$lastName" },

totalSales: { $sum: { $multiply: ["$orderDetails.quantity", "$orderDetails.unitPrice"] } }

}

},

{

$sort: {

totalSales: -1

}

},

{

$limit: 1

},

{

$project: {

\_id: 0,

firstName: "$\_id.firstName",

lastName: "$\_id.lastName",

totalSales: 1

}

}

]);

1. Sql :

SELECT p.productID, p.productName, s.supplierName, c.categoryName AS productCategory FROM products p JOIN suppliers s ON p.supplierID = s.supplierID JOIN categories c ON p.categoryID = c.categoryID ORDER BY c.categoryName;

MongoDB :  
db.products.aggregate([

{

$lookup: {

from: "suppliers",

localField: "supplierID",

foreignField: "supplierID",

as: "supplier"

}

},

{

$unwind: "$supplier"

},

{

$lookup: {

from: "categories",

localField: "categoryID",

foreignField: "categoryID",

as: "category"

}

},

{

$unwind: "$category"

},

{

$project: {

\_id: 0,

productID: 1,

productName: 1,

supplierName: "$supplier.supplierName",

productCategory: "$category.categoryName"

}

},

{

$sort: {

productCategory: 1 } } ]);

1. SQL :

SELECT r.regionDescription, COUNT(e.employeeID) AS employeeCount FROM employees e JOIN territories t ON e.employeeID = t.employeeID JOIN region r ON t.regionID = r.regionID GROUP BY r.regionDescription;

MongoDB :

db.employees.aggregate([

{

$lookup: {

from: "territories",

localField: "employeeID",

foreignField: "employeeID",

as: "territories"

}

},

{

$unwind: "$territories"

},

{

$lookup: {

from: "regions",

localField: "territories.RegionID",

foreignField: "regionID",

as: "region"

}

},

{

$unwind: "$region"

},

{

$group: {

\_id: "$region.regionDescription",

EmployeeCount: { $sum: 1 }

}

}

]);

1. Sql :

SELECT r.regionDescription, SUM(od.quantity \* od.unitPrice) AS salesAmount FROM region r JOIN territories t ON r.regionID = t.regionID JOIN employees e ON t.employeeID = e.employeeID JOIN orders o ON e.employeeID = o.employeeID JOIN order\_details od ON o.orderID = od.orderID GROUP BY r.regionDescription;

MongoDB :

db.region.aggregate([

{

$lookup: {

from: "territories",

localField: "regionID",

foreignField: "regionID",

as: "territories"

}

},

{

$unwind: "$territories"

},

{

$lookup: {

from: "employees",

localField: "territories.employeeID",

foreignField: “employeeID",

as: "employees"

}

},

{

$unwind: "$employees"

},

{

$lookup: {

from: "orders",

localField: "employees.employeeID",

foreignField: "employeeID",

as: "orders"

}

},

{

$unwind: "$orders"

},

{

$lookup: {

from: "order\_details",

localField: "orders.orderID",

foreignField: "orderID",

as: "orderDetails"

}

},

{

$unwind: "$orderDetails"

},

{

$group: {

\_id: "$regionDescription",

salesAmount: { $sum: { $multiply: ["$orderDetails.quantity", "$orderDetails.unitPrice"] } }

}

}

]);

1. Sql :

SELECT AVG(total) AS averageOrderValue

FROM (

SELECT SUM(od.quantity \* od.unitPrice) AS total

FROM orders o

JOIN order\_details od ON o.orderID = od.orderID

GROUP BY o.orderID

) AS subquery;

MongoDB :

db.orders.aggregate([

{

$lookup: {

from: "order\_details",

localField: "orderID",

foreignField: "orderID",

as: "orderDetails"

}

},

{

$unwind: "$orderDetails"

},

{

$group: {

\_id: "$orderID",

total: { $sum: { $multiply: ["$orderDetails.quantity", "$orderDetails.unitPrice"] } }

}

},

{

$group: {

\_id: null,

averageOrderValue: { $avg: "$total" }

}

}

]);

1. SQL :

SELECT o.orderID, o.orderDate, c.customerName

FROM orders o

JOIN customers c ON o.customerID = c.customerID

JOIN (

SELECT AVG(total) AS averageOrderValue

FROM (

SELECT o.orderID, SUM(od.quantity \* od.unitPrice) AS total

FROM orders o

JOIN order\_details od ON o.orderID = od.orderID

GROUP BY o.orderID

) AS subquery

) AS avg ON (SELECT SUM(od.quantity \* od.unitPrice) FROM order\_details od WHERE od.orderID = o.orderID) > avg.averageOrderValue;

MongoDB :

db.orders.aggregate([

{

$lookup: {

from: "customers",

localField: "customerID",

foreignField: "customerID",

as: "customer"

}

},

{

$unwind: "$customer"

},

{

$lookup: {

from: "order\_details",

localField: "orderID",

foreignField: "orderID",

as: "orderDetails"

}

},

{

$unwind: "$orderDetails"

},

{

$group: {

\_id: "$orderID",

OrderDate: { $first: "$orderDate" },

CustomerName: { $first: "$customer.customerName" },

total: { $sum: { $multiply: ["$orderDetails.quantity", "$orderDetails.unitPrice"] } }

}

},

{

$lookup: {

from: "orders",

let: { orderId: "$\_id" },

pipeline: [

{

$lookup: {

from: "order\_details",

localField: "orderID",

foreignField: "orderID",

as: "orderDetails"

}

},

{

$unwind: "$orderDetails"

},

{

$group: {

\_id: "$orderID",

averageOrderValue: { $avg: { $multiply: ["$orderDetails.quantity", "$orderDetails.unitPrice"] } }

}

}

],

as: "avg"

}

},

{

$unwind: "$avg"

},

{

$match: {

total: { $gt: "$avg.averageOrderValue" }

}

},

{

$project: {

\_id: 0,

orderID: "$\_id",

orderDate: 1,

customerName: 1

}

}

]);

1. SQL :

SELECT c.customerID, c.customerName, COALESCE(SUM(od.quantity \* od.unitPrice), 0) AS totalSales FROM customers c LEFT JOIN orders o ON c.customerID = o.customerID LEFT JOIN order\_details od ON o.orderID = od.orderID GROUP BY c.customerID, c.customerName;

MongoDB:

db.customers.aggregate([

{

$lookup: {

from: "orders",

localField: "customerID",

foreignField: "customerID",

as: "orders"

}

},

{

$unwind: { path: "$orders", preserveNullAndEmptyArrays: true }

},

{

$lookup: {

from: "order\_details",

localField: "orders.orderID",

foreignField: "orderID",

as: "orderDetails"

}

},

{

$unwind: { path: "$orderDetails", preserveNullAndEmptyArrays: true }

},

{

$group: {

\_id: {

CustomerID: "$customerID",

CustomerName: "$customerName"

},

totalSales: { $sum: { $multiply: [{ $ifNull: ["$orderDetails.quantity", 0] }, { $ifNull: ["$orderDetails.unitPrice", 0] }] } }

}

},

{

$project: {

\_id: 0,

CustomerID: "$\_id.customerID",

CustomerName: "$\_id.customerName",

totalSales: 1

}

}

]);

1. SQL :

SELECT productID, productName, unitsInStock, reorderLevel

FROM products

WHERE unitsInStock <= reorderLevel AND discontinued = 0;

MongoDB :

db.products.find({

unitsInStock: { $lte: "$reorderLevel" },

discontinued: { $eq: false }

}, {

productID: 1,

productName: 1,

unitsInStock: 1,

reorderLevel: 1

});