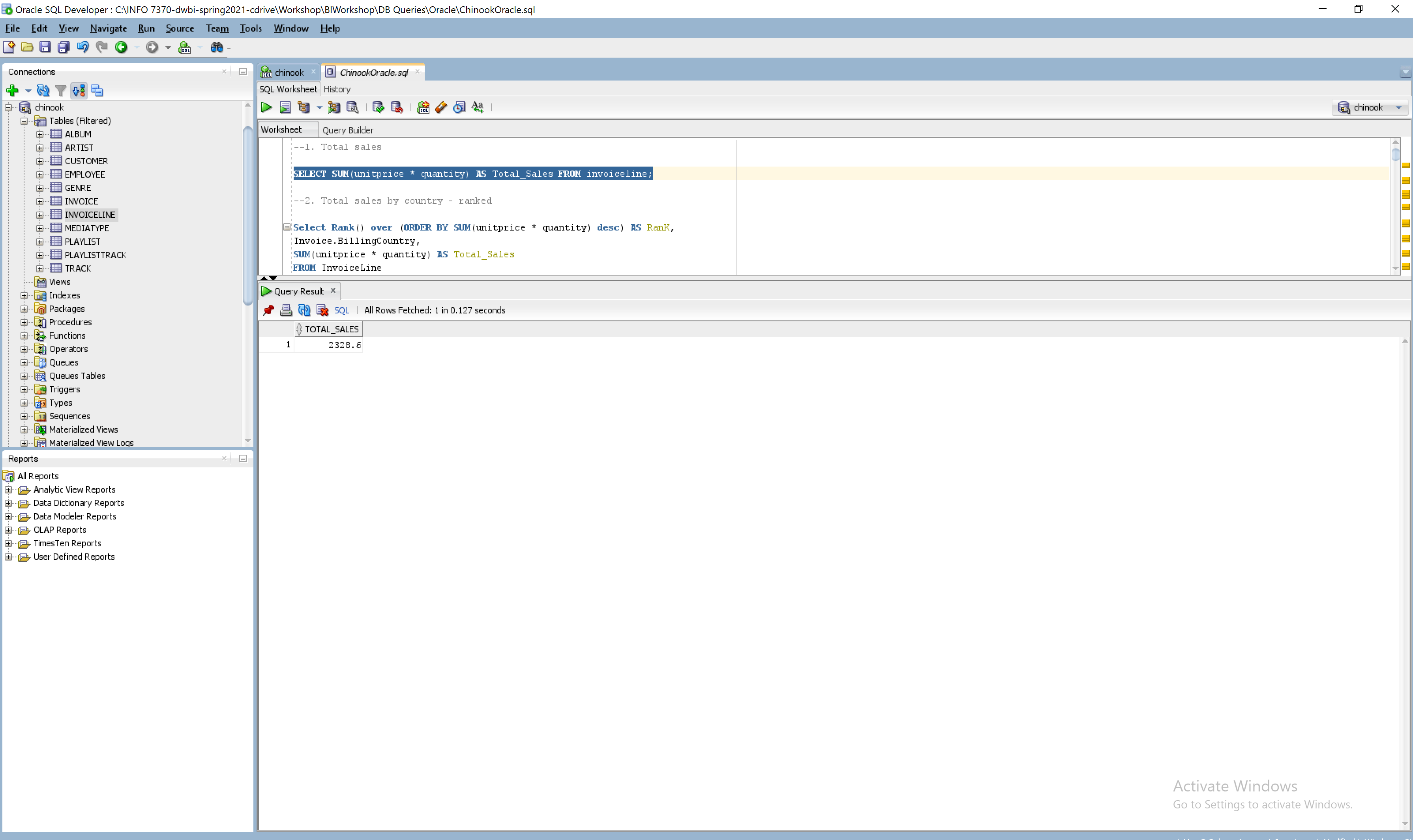
**Oracle Queries Screen Shot:**

**1.Total sales $ -   
Query:**SELECT SUM(unitprice \* quantity) AS Total\_Sales FROM invoiceline; ****

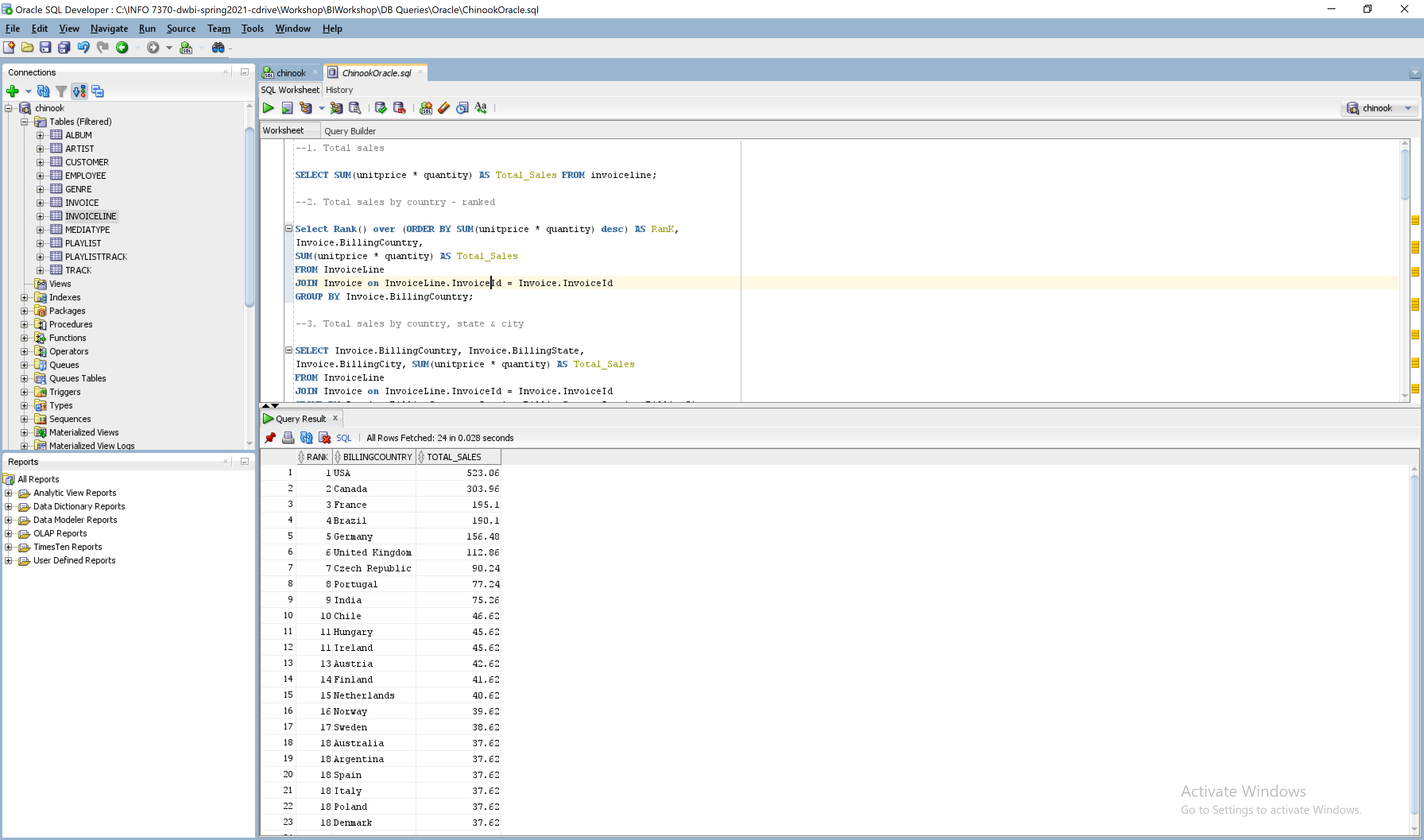
**2.Total sales $ by country –ranked (or at least sorted largest to smallest) -**Select Rank() over (ORDER BY SUM(unitprice \* quantity) desc) AS RanK,

Invoice.BillingCountry,

SUM(unitprice \* quantity) AS Total\_Sales

FROM InvoiceLine

JOIN Invoice on InvoiceLine.InvoiceId = Invoice.InvoiceId

GROUP BY Invoice.BillingCountry; ****

**3.Total sales $ by country, state & city -   
Query:**SELECT Invoice.BillingCountry, Invoice.BillingState,

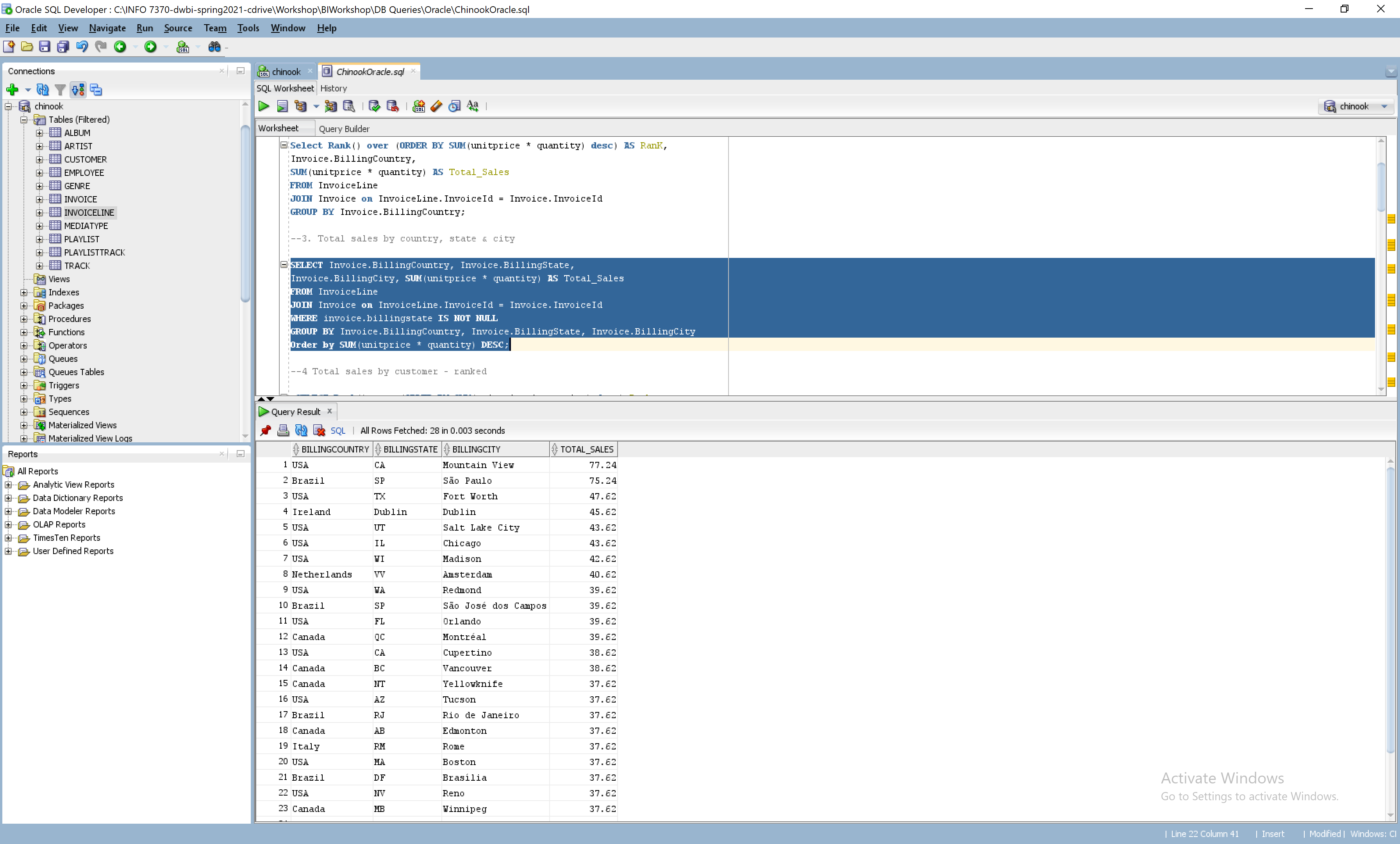
Invoice.BillingCity, SUM(unitprice \* quantity) AS Total\_Sales

FROM InvoiceLine

JOIN Invoice on InvoiceLine.InvoiceId = Invoice.InvoiceId

WHERE invoice.billingstate IS NOT NULL

GROUP BY Invoice.BillingCountry, Invoice.BillingState, Invoice.BillingCity

Order by SUM(unitprice \* quantity) DESC;

**4.Total sales $ by customer (a person with last name & first name) –ranked (or at least sorted largest to smallest)  
Query:**

SELECT Rank() over (ORDER BY SUM(unitprice \* quantity) desc) Rank,

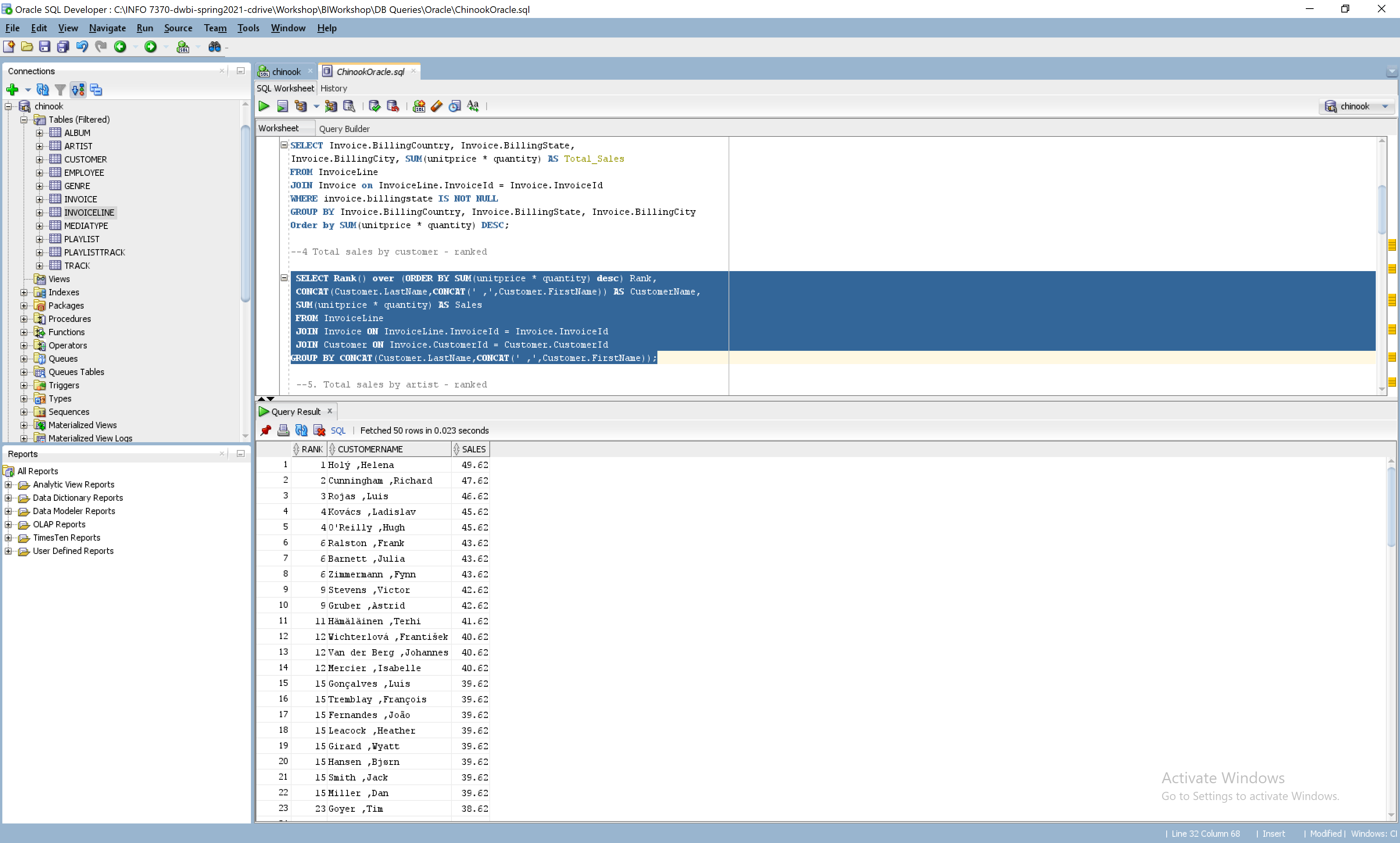
CONCAT(Customer.LastName,CONCAT(' ,',Customer.FirstName)) AS CustomerName,

SUM(unitprice \* quantity) AS Sales

FROM InvoiceLine

JOIN Invoice ON InvoiceLine.InvoiceId = Invoice.InvoiceId

JOIN Customer ON Invoice.CustomerId = Customer.CustomerId

GROUP BY CONCAT(Customer.LastName,CONCAT(' ,',Customer.FirstName));

**5.Total sales $ by artist –ranked (or at least sorted largest to smallest)  
Query:**

SELECT RANK() over (ORDER BY SUM(Invoiceline.UnitPrice \* Quantity) DESC) Rank,

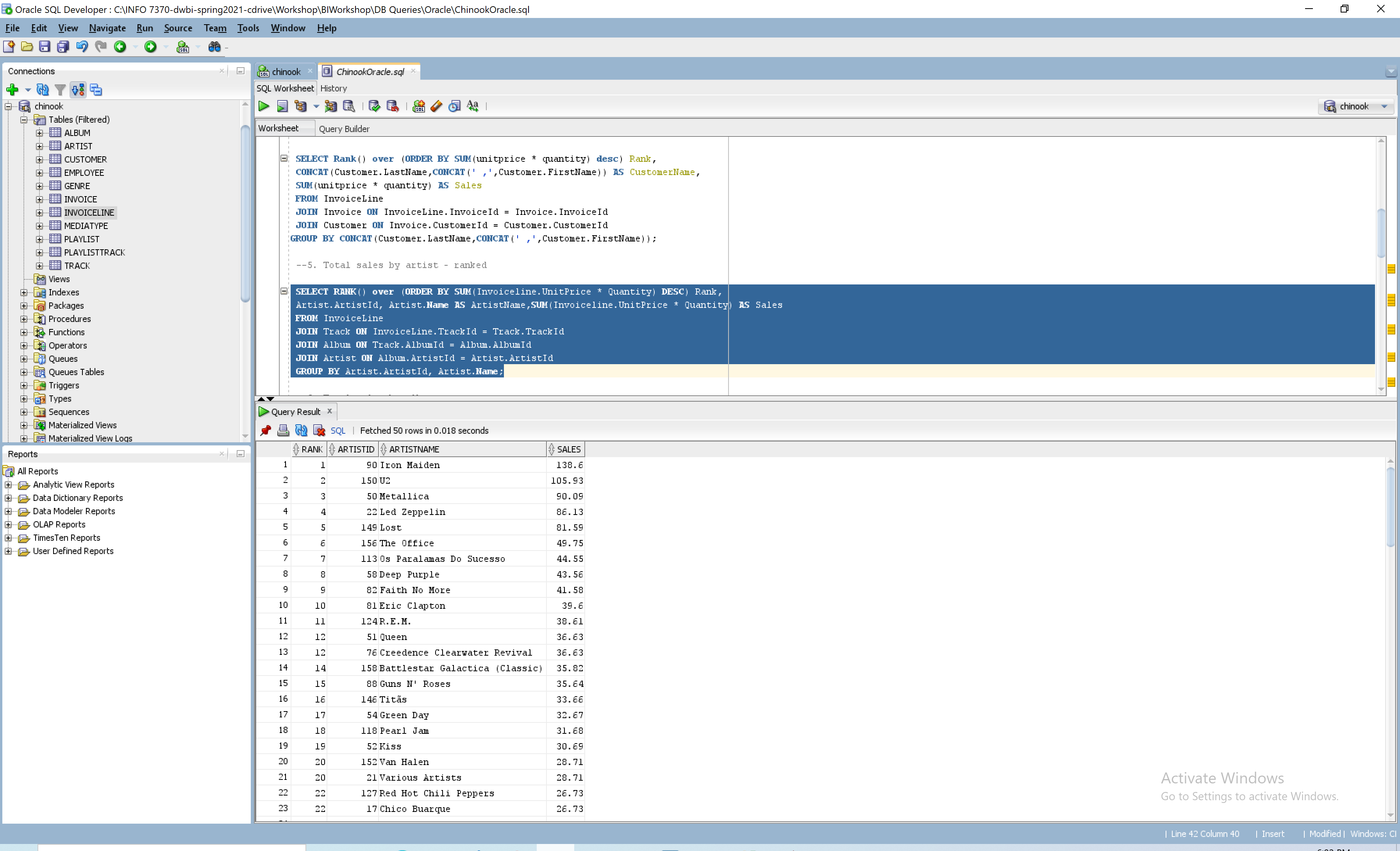
Artist.ArtistId, Artist.Name AS ArtistName,SUM(Invoiceline.UnitPrice \* Quantity) AS Sales

FROM InvoiceLine

JOIN Track ON InvoiceLine.TrackId = Track.TrackId

JOIN Album ON Track.AlbumId = Album.AlbumId

JOIN Artist ON Album.ArtistId = Artist.ArtistId

GROUP BY Artist.ArtistId, Artist.Name;

**6.Total sales $ by albums  
Query:**

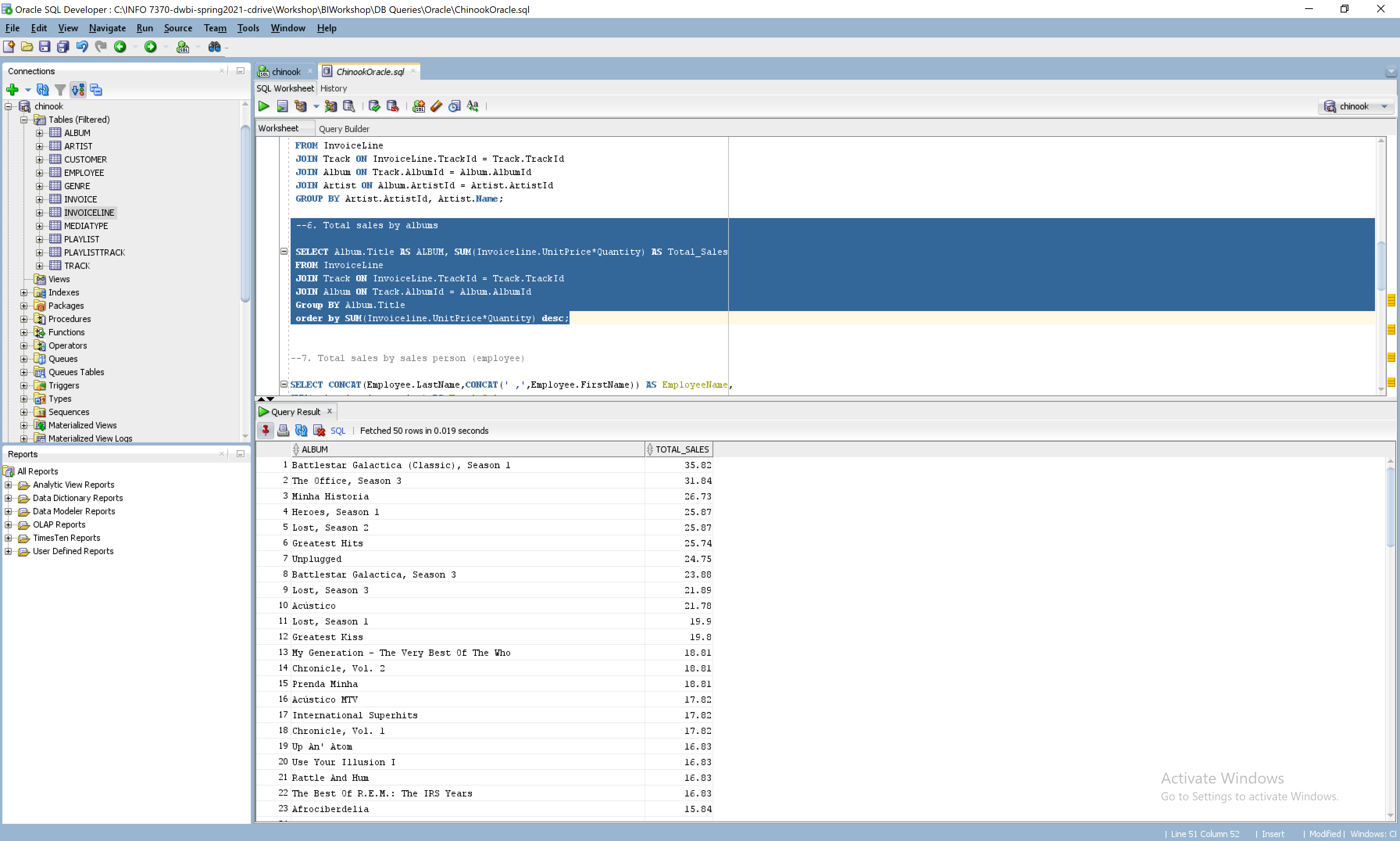
SELECT Album.Title AS ALBUM, SUM(Invoiceline.UnitPrice\*Quantity) AS Total\_Sales

FROM InvoiceLine

JOIN Track ON InvoiceLine.TrackId = Track.TrackId

JOIN Album ON Track.AlbumId = Album.AlbumId

Group BY Album.Title

order by SUM(Invoiceline.UnitPrice\*Quantity) desc;  


**7.Total sales $ by salesperson (employee)  
Query:**

SELECT CONCAT(Employee.LastName,CONCAT(' ,',Employee.FirstName)) AS EmployeeName,

SUM(unitprice \* quantity) AS Total\_Sales

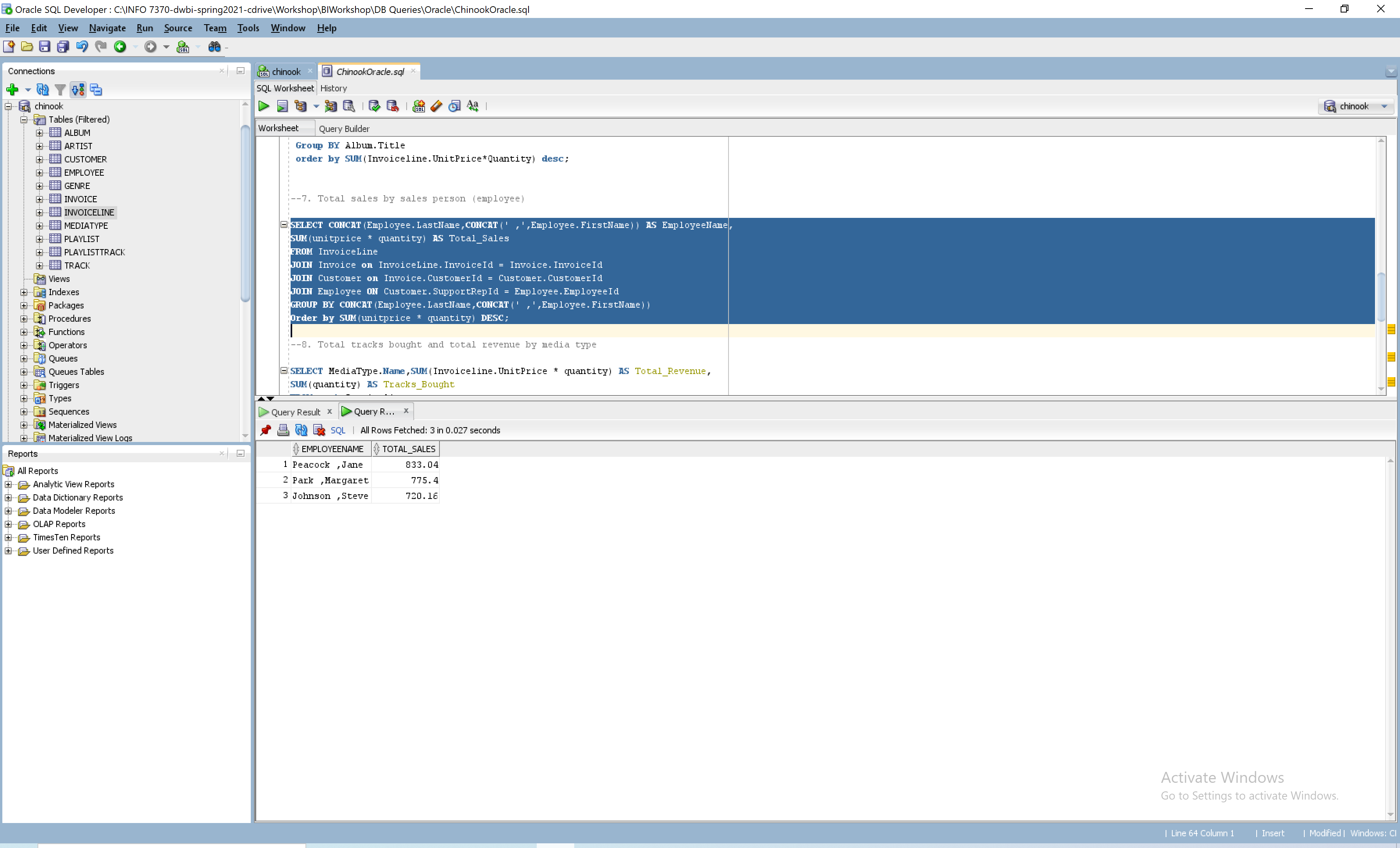
FROM InvoiceLine

JOIN Invoice on InvoiceLine.InvoiceId = Invoice.InvoiceId

JOIN Customer on Invoice.CustomerId = Customer.CustomerId

JOIN Employee ON Customer.SupportRepId = Employee.EmployeeId

GROUP BY CONCAT(Employee.LastName,CONCAT(' ,',Employee.FirstName))

Order by SUM(unitprice \* quantity) DESC;

**8.Total tracks bought and total revenue $ by media type  
Query:**SELECT MediaType.Name,SUM(Invoiceline.UnitPrice \* quantity) AS Total\_Revenue,

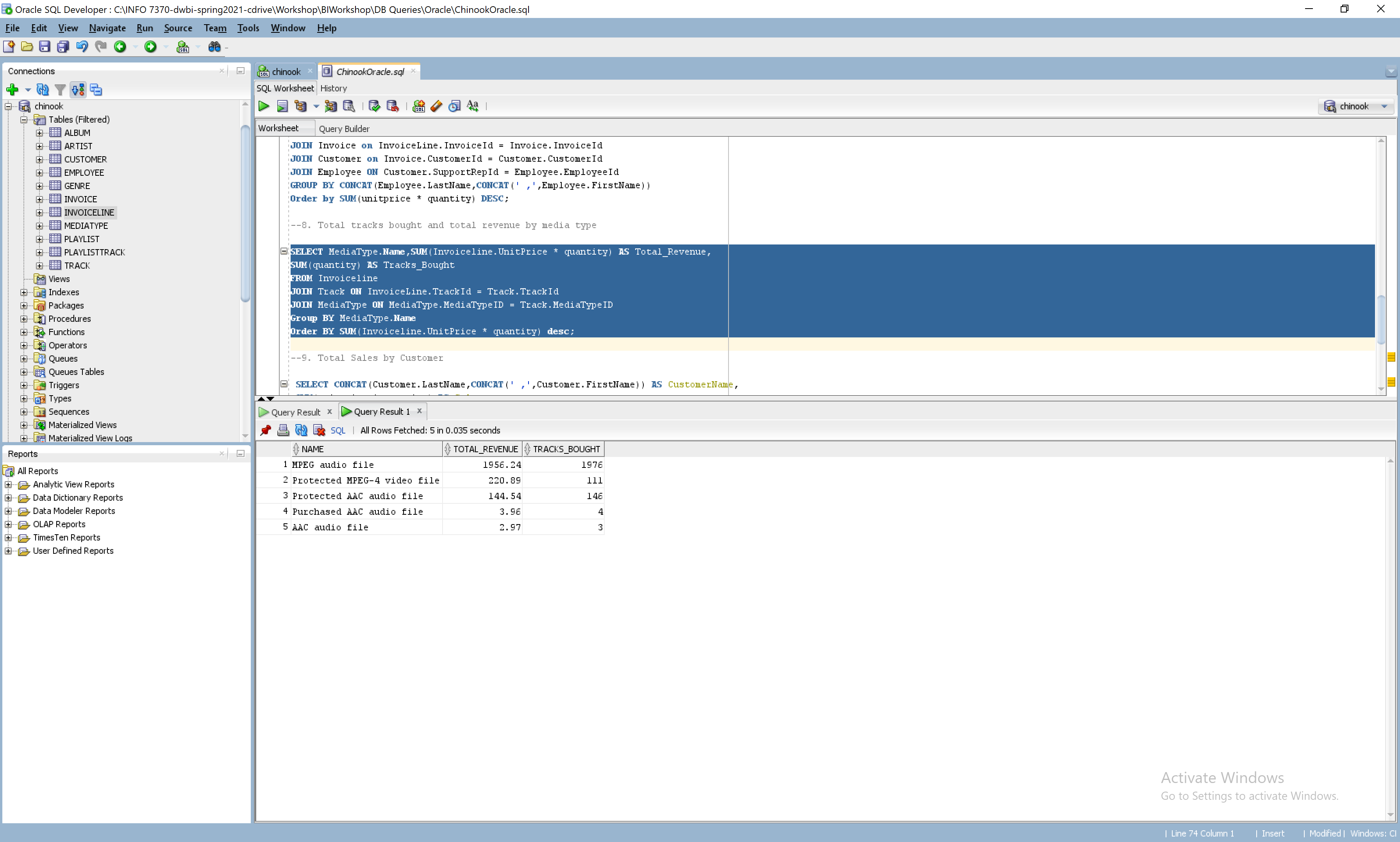
SUM(quantity) AS Tracks\_Bought

FROM Invoiceline

JOIN Track ON InvoiceLine.TrackId = Track.TrackId

JOIN MediaType ON MediaType.MediaTypeID = Track.MediaTypeID

Group BY MediaType.Name

Order BY SUM(Invoiceline.UnitPrice \* quantity) desc;

**9.Total sales $ by genre  
Query:**

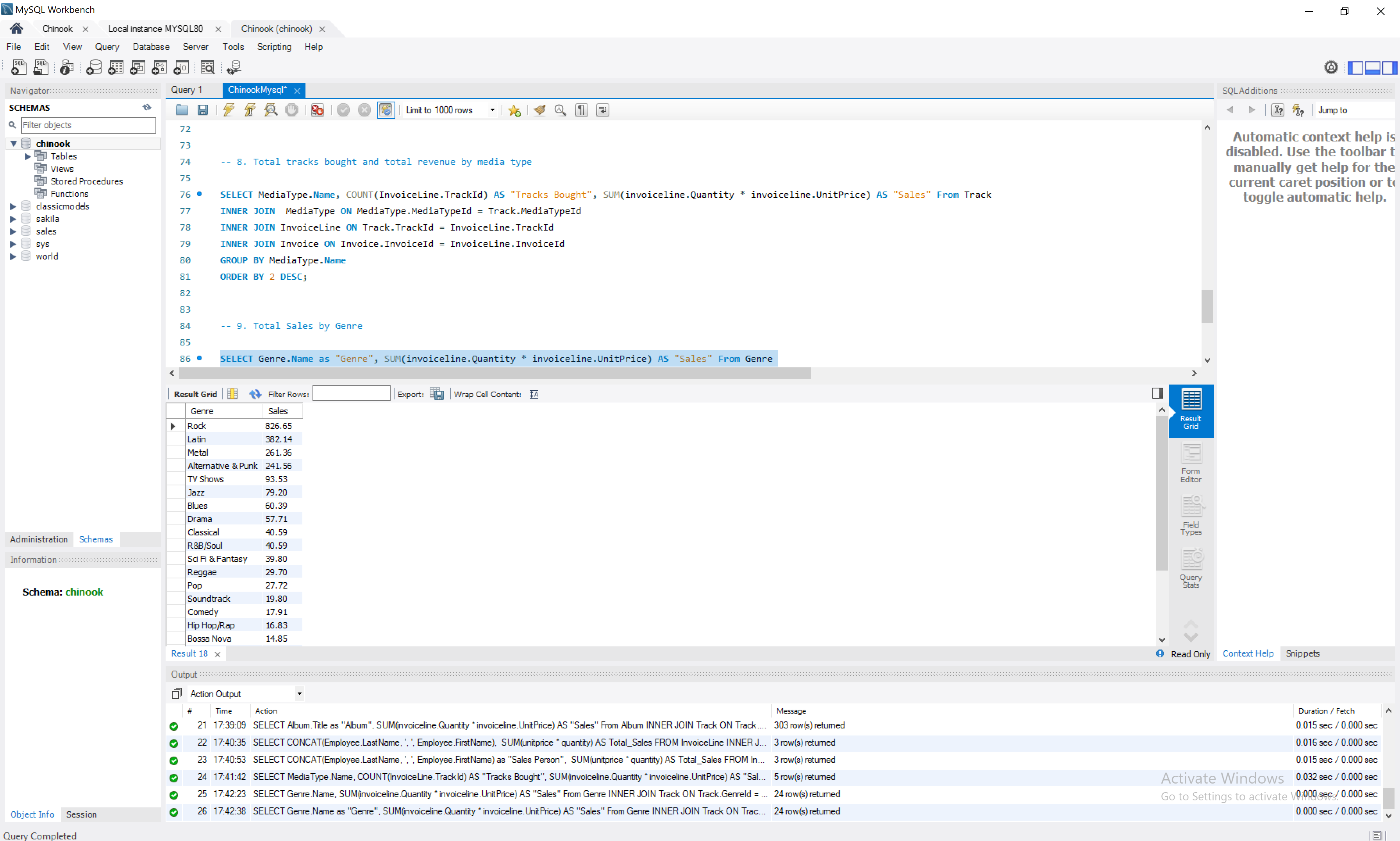
SELECT Genre.Name as "Genre", SUM(invoiceline.Quantity \* invoiceline.UnitPrice) AS "Sales" From Genre

INNER JOIN Track ON Track.GenreId = GENRE.GenreId

INNER JOIN InvoiceLine ON InvoiceLine.TrackId = Track.TrackId

INNER JOIN Invoice ON Invoice.InvoiceId = InvoiceLine.InvoiceId

Group by Genre.Name

ORDER BY 2 DESC; ****

**10.Total sales $ by company**

**Query:**

SELECT CONCAT(Customer.LastName,CONCAT(' ,',Customer.FirstName)) AS CustomerName,Customer.Company,

SUM(unitprice \* quantity) AS Sales

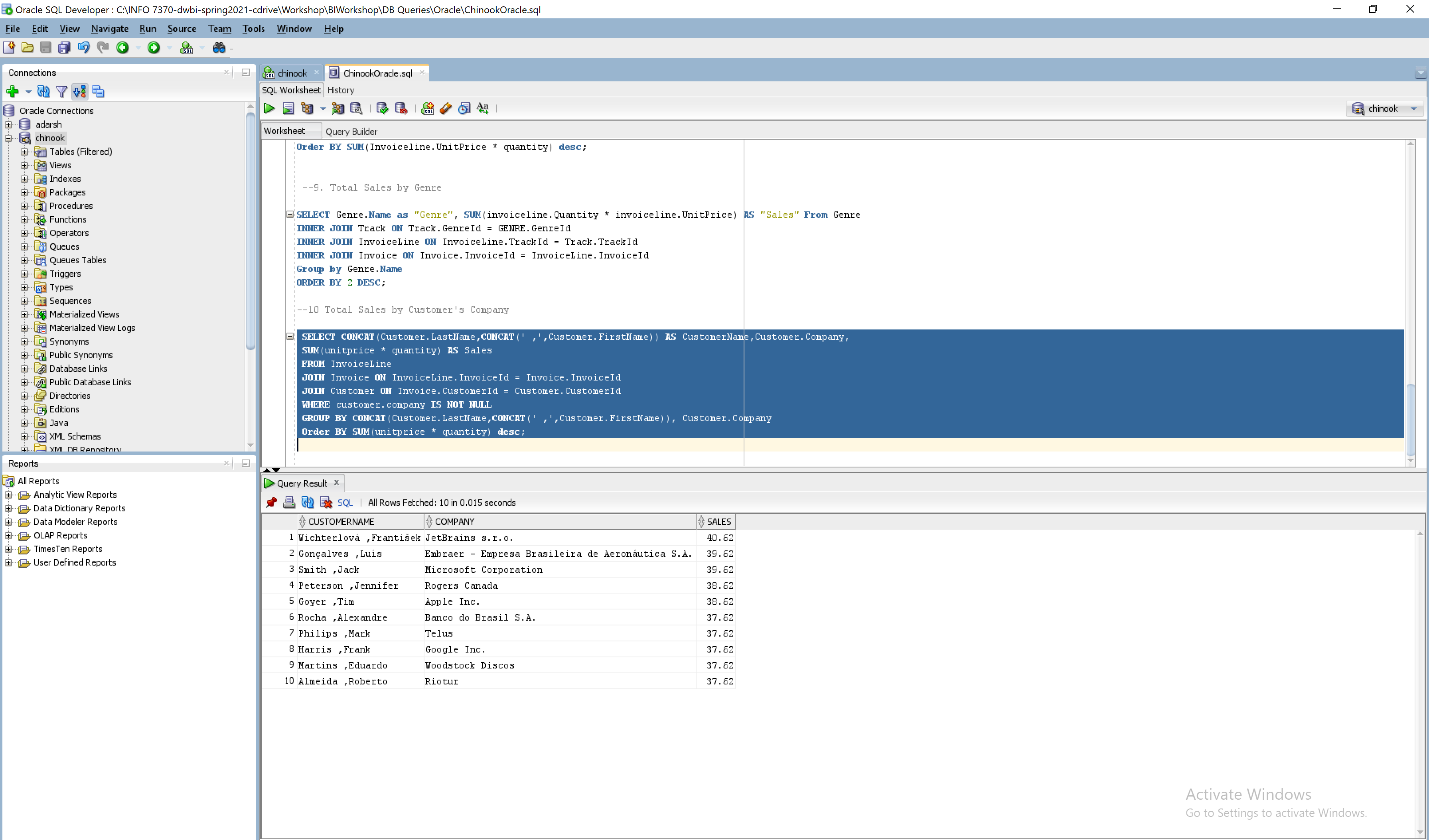
FROM InvoiceLine

JOIN Invoice ON InvoiceLine.InvoiceId = Invoice.InvoiceId

JOIN Customer ON Invoice.CustomerId = Customer.CustomerId

WHERE customer.company IS NOT NULL

GROUP BY CONCAT(Customer.LastName,CONCAT(' ,',Customer.FirstName)), Customer.Company

Order BY SUM(unitprice \* quantity) desc; ****