

DESIGNING ADVANCED DATA ARCHITECTURE FOR BUSINESS INTELLIGENCE

ASSIGNMENT 2 – MySQL queries and outputs

00214365

1. Total sales \$ via Invoice

The screenshot shows the MySQL Workbench interface with the following details:

- Query Editor:** Contains the following SQL code:

```
1 show databases;
2 use chinook;
3 select sum(total) as Sales_via_invoice from invoice;
```
- Result Grid:** Displays the result of the query: **Sales_via_invoice** with a value of **2328.60**.
- Action Output:** Shows the history of actions taken during the session, including the execution of the three statements above.
- Object Info:** Shows that no object is selected.

2. Total sales \$ via Invoiceline

The screenshot shows the MySQL Workbench interface with the following details:

- Object Browser:** Shows the schema structure, specifically the **Employee** table and its columns (**EmployeeId**, **LastName**, **FirstName**, **Title**, **ReportsTo**, **BirthDate**, **HireDate**, **Address**, **City**, **State**, **Country**, **PostalCode**, **Phone**, **Fax**, **Email**).
- Query Editor:** Contains the following SQL code:

```
select sum(unitprice*quantity) as TotalSales_via_InvoiceLine from invoiceLine;
```
- Result Grid:** Displays the result of the query: **TotalSales_via_InvoiceLine** with a value of **2328.60**.
- Action Output:** Shows the history of actions taken during the session, including the execution of the query above.
- Object Info:** Shows that no object is selected.

DESIGNING ADVANCED DATA ARCHITECTURE FOR BUSINESS INTELLIGENCE

ASSIGNMENT 2 – MySQL queries and outputs

00214365

3. Total tracks (songs) sold

The screenshot shows the MySQL Workbench interface with a query editor and a result grid. The query is:

```
1 show databases;
2 use chinook;
3 select sum(total) as Sales_via_Invoice from invoice;
4 select sum(unitprice) as TotalSales_via_InvoiceLine from invoiceLine;
5 select count(trackid) as Tracks_Sold from invoiceLine;
```

The result grid displays a single row with the value 2240 for 'Tracks_Sold'.

Below the result grid is a timeline showing the execution of each query step, with details like time, action, response, and duration/fetch time.

Time	Action	Response	Duration / Fetch Time
5 08:32:27	select sum(invoiceLine.UnitPrice) as tot, Artist.Name from invoice join invoiceLine on invoice.InvoiceId=invoiceLine.InvoiceId;	165 row(s) returned 0 row(s) affected	0.037 sec / 0.00028...
6 18:47:37	use chinook;		0.020 sec
7 18:48:26	select sum(total) from invoice LIMIT 0, 1000	1 row(s) returned	0.026 sec / 0.00020...
8 18:49:04	select sum(total) as Sales_via_Invoice from invoice LIMIT 0, 1000	1 row(s) returned	0.0022 sec / 0.0001...
9 18:50:29	select sum(unitprice) as TotalSales_via_InvoiceLine from invoiceLine LIMIT 0, 1000	1 row(s) returned	0.0047 sec / 0.0001...
10 18:51:14	select * from track LIMIT 0, 1000	1000 row(s) returned	0.0051 sec / 0.0025 ...
11 18:51:57	select * from invoiceLine LIMIT 0, 1000	1000 row(s) returned	0.0030 sec / 0.0004...
12 18:52:17	select count(trackid) from invoiceLine LIMIT 0, 1000	1 row(s) returned	0.0076 sec / 0.0001...
13 18:52:32	select count(trackid) as Tracks_Sold from invoiceLine LIMIT 0, 1000	1 row(s) returned	0.0016 sec / 0.0001...

Query Completed

4. Total sales \$ by customer's country – ranked (sorted largest to smallest)

The screenshot shows the MySQL Workbench interface with a query editor and a result grid. The query is:

```
9 select concat(customer.LastName, ' ', customer.FirstName) as Customer_Name, customer.country,
10 sum(total) as Total_Sales from invoice
11 join customer on invoice.customerId=customer.customerId group by Customer_Name, customer.country, state,
12 customer.city order by Total_Sales desc;
```

The result grid displays a list of customers with their total sales, ordered by total sales in descending order. The columns are 'Customer_Name', 'country', and 'Total_Sales'.

Customer_Name	country	Total_Sales
Holy,Helena	Czech Republic	49.62
Cunningham,Richard	USA	47.62
Rojas,Maria	Cuba	45.62
O'Reilly,Agh	Ireland	45.62
Kovács,Ladislav	Hungary	45.62
Zimmermann,Fynn	Germany	43.62
Raleston,Frank	USA	43.62
Burman,Paul	USA	43.62
Gruber,Ariad	Austria	42.62
Stevens,Victor	USA	42.62
Hilmäläinen,Terhi	Finland	41.62
Wichterlák,Franšíšek	Czech Republic	40.62
Van,Brian	United Kingdom	40.62
Mercier,Iabelle	France	40.62
Gongalves,Luis	Brazil	39.62
Tremblay,François	Canada	39.62
Hansen,Sjørn	Norway	39.62
Orsini,Francesca	France	39.62
Smith,Jack	USA	39.62
Miller,Dan	USA	39.62
Leacock,Heather	USA	39.62
Petersen,André	Portugal	38.62
Parsons,Karen	USA	38.62
Johansson,Joakim	Sweden	38.62
Peterson,Jennifer	Canada	38.62
Lefebvre,Dominique	France	38.62

Below the result grid is a timeline showing the execution of each query step, with details like time, action, response, and duration/fetch time.

Time	Action	Response	Duration / Fetch Time
204 22:19:07	select concat(employee.LastName, ' ', employee.FirstName) as Employee_Name, sum(case when ((TIMESTAMPDIFF(...))>0) then 1 else 0 end) as Total_Sales from employee;	3 row(s) returned	UUUJI sec / UUUUUU...
205 22:20:47	select concat(employee.LastName, ' ', employee.FirstName) as Employee_Name, sum(case when ((TIMESTAMPDIFF(...))>0) then 1 else 0 end) as Total_Sales from employee;	3 row(s) returned	0.0070 sec / 0.0001...
206 22:21:51	select concat(employee.LastName, ' ', employee.FirstName) as Employee_Name, sum(case when ((TIMESTAMPDIFF(...))>0) then 1 else 0 end) as Total_Sales from employee;	3 row(s) returned	0.0056 sec / 0.0001...
207 23:03:13	select concat(concat(customer.LastName, ' ', customer.FirstName) as Customer_Name, customer.country, sum(total) as Tot...)	59 row(s) returned	0.053 sec / 0.00062...

Query Completed

DESIGNING ADVANCED DATA ARCHITECTURE FOR BUSINESS INTELLIGENCE

ASSIGNMENT 2 – MySQL queries and outputs

00214365

5. Total sales \$ by customer's geo (country, state & city)

The screenshot shows the SSMS interface with the following details:

- Object Explorer (Left):** Shows the database structure with nodes like Schemas, Customer, Columns, Indexes, Foreign Keys, Triggers, Employee, Genre, and so on.
- Query Editor (Top):** Contains a query window titled "Query 1" with the following T-SQL code:

```
select customer.country, ifnull(customer.state,'Not Specified') as state,customer.city, sum(total) as Total_Sales from invoice
join customer on invoice.customerId=customer.customerId group by customer.country, state, customer.city order by Total_Sales desc;
```
- Result Grid (Bottom):** Displays the results of the query as a table with columns: country, state, city, and Total_Sales. The data includes rows for various countries and cities like Prague, Mountain View, Paris, London, São Paulo, Berlin, Fort Worth, Santiago, Dublin, Budapest, Louisville, Salt Lake City, Frankfurt, Vienna, Madison, Helsinki, Amsterdam, Dhaka, São José do Rio Preto, Montreal, Oslo, Redmond, Ciudad de México, Bordeaux, Lisbon, Delhi, Stockholm, Vancouver, Calgary, Coquitlam, Yellowknife, Winnipeg, Porto, Halifax, Ottawa, and Toronto.
- Toolbar (Top):** Includes icons for Home, Administration, Schemas, Tables, Views, Procedures, Functions, Scripts, and Logins.
- Status Bar (Bottom):** Shows "Query Completed" and "Result 14".
- Right Sidebar:** Features sections for Result Grid, Form Editor, Field Types, Query Stats, and Execution Plan.

6. Total sales \$ by customer (a person with last name & first name) – ranked (sorted largest to smallest)

The screenshot shows the MySQL Workbench interface with the following details:

- Top Bar:** Local instance 3306, Administration, Schemas, Query 1 (sakila-data*), SQL File 4*, Limit to 1000 rows.
- Schemas List:** Tables, Album, Artist, Customer (selected), Columns.
- Query Editor:** SQL code:

```
select concat(customer.LastName, ' ', customer.FirstName) as Customer_Name, sum(invoice.total) as Total_Sales from customer
join invoice on customer.customerId=invoice.customerId
group by Customer_Name order by Total_Sales desc;
```
- Result Grid:** Customer_Name (Column 1), Total_Sales (Column 2). The results list 100 rows of customer names and their total sales, such as Helga Helgstrand (49.62), Richard Cunningham (47.62), Luis Rojas (46.62), and others.
- Right Sidebar:** Form Editor, Field Types, Query Stats, Execution Plan.
- Bottom Status Bar:** Result 16, Read Only.

DESIGNING ADVANCED DATA ARCHITECTURE FOR BUSINESS INTELLIGENCE

ASSIGNMENT 2 – MySQL queries and outputs

00214365

7. Total sales \$ by company – ranked (sorted largest to smallest)

The screenshot shows the MySQL Workbench interface with a query editor and results grid. The query selects total sales from the customer table, grouping by company and ordering by total sales in descending order. The results show various companies and their total sales.

Company	Total_Sales
Others	1943.40
jetbrains s.r.o.	40.62
Embraer - Empresa Brasileira de Aeronáutica S.A.	39.62
Microsoft Corporation	38.62
Apple Inc.	38.62
Woodstock Discos	37.62
Banco do Brasil S.A.	37.62
Riotur	37.62
Telus	37.62
Google Inc.	37.62

8. Total sales \$ by artist and top 20 artists– ranked (sorted largest to smallest)

The screenshot shows the MySQL Workbench interface with a query editor and results grid. The query selects total sales from the artist table, joining it with album, track, and invoiceLine tables to get the total sales per artist. The results show the top 20 artists with their total sales.

Artist_Name	Total_Sales
Iron Maiden	138.60
U2	105.93
Metallica	90.09
Led Zeppelin	86.13
Lost	81.59
The Office	49.71
Os Mutantes Do Sucesso	45.45
Deep Purple	43.56
Faith No More	41.58
Eric Clapton	39.60
R.E.M.	38.61
Queen	36.96
Unchor Cisnerow R...	35.93
Battlestar Galactica (Gas...	35.82
Guns N' Roses	35.64
Titus	33.66
Green Day	32.60
Pearl Jam	31.58
Kiss	30.69
Various Artists	28.71
Van Halen	28.71
Chico Buarque	26.73
Red Hot Chili Peppers	25.70
Hootie & the Blowfish	25.47
Lenny Kravitz	25.74
Chico Science & Nacio Z...	24.75
Battlestar Galactica	23.88
Ozzy Osbourne	23.76
Smashing Pumpkins	23.76

DESIGNING ADVANCED DATA ARCHITECTURE FOR BUSINESS INTELLIGENCE

ASSIGNMENT 2 – MySQL queries and outputs

00214365

```

Local Instance 3306
Administration Schemas Query 1 sakila-data* SQL File 4*
SCHEMAS Filter objects Limit to 1000 rows
Customer Employee
Employee Columns
Artist_Name Total_Sales
Iron Maiden 138.60
U2 105.93
Metallica 90.09
Led Zeppelin 86.13
Lost 81.59
The Office 74.84
Os Paralamas Do Sucesso 44.65
Deep Purple 43.56
Faith No More 41.58
Eric Clapton 39.60
Result 153
Action Output
Time Action Response Duration / Fetch Time
182 21:40:02 select ifNull(artist.Name, 'Unknown') as Artist_Name, sum(invoiceLine.unitprice*invoiceLine.quantity) as Total_Sales from artist join album on artist.artistId=album.artistId join track on album.albumId=track.albumId join invoiceLine on track.trackId=invoiceLine.trackId group by Artist_Name order by Total_Sales desc limit 10;
0.024 sec / 0.0000017...
Query Completed
  
```

9. Total sales \$ by album – ranked (sorted largest to smallest)

```

Local Instance 3306
Administration Schemas Query 1 sakila-data* SQL File 4*
SCHEMAS Filter objects Limit to 1000 rows
Customer Employee
Employee Columns
Album_Name Total_Sales
Battlestar Galactica (Classic), Season 1 35.82
Minha Minha 34.65
The Office, Season 3 31.84
Heroes, Season 1 25.67
Lost, Season 2 25.67
Greatest Hits 24.75
Lost, Season 4 24.75
Battlestar Galactica, Season 3 23.88
Lost, Season 3 21.89
Acústico 21.78
Lost, Season 1 21.78
Greatest Hits 19.90
Prenda Minha 18.81
Chronicle, Vol. 2 18.81
My Generation - The Very Best Of The... 18.81
International Superhits 17.82
Chronicle, Vol. 1 17.82
Acústico MTV 17.82
Up An Atom 16.83
Use Your Illusion I 16.83
The Best Of R.E.M.: The IRS Years 16.83
Hello And Hum 16.55
Alcoolicos 16.44
Instant Karma: The Amnesty Internatio... 15.84
Chill: Brazil (Disc 2) 14.85
Vinicio De Moraes - Sem Limite 14.85
Eric Clapton 14.85
American Idol 14.85
Armin Van Os Paralamas Do Sucesso 14.60
Result 154
Action Output
Time Action Response Duration / Fetch Time
183 21:41:21 select ifNull(album.Title, 'Unknown') as Album_Name, sum(invoiceLine.unitprice*invoiceLine.quantity) as Total_Sales from album join invoiceLine on album.albumId=invoiceLine.albumId group by Album_Name order by Total_Sales desc;
0.014 sec / 0.000007...
Query Completed
  
```

DESIGNING ADVANCED DATA ARCHITECTURE FOR BUSINESS INTELLIGENCE

ASSIGNMENT 2 – MySQL queries and outputs

00214365

10. Total sales \$ by salesperson (employee)

```

    select concat(employee.LastName, ' ', employee.FirstName) as Employee_Name, sum(total) as Total_Sales from employee
    join customer on employee.employeeId=customer.supportRepId
    join invoice on customer.customerId = invoice.customerId
    group by Employee_Name;
  
```

The screenshot shows the MySQL Workbench interface with the following details:

- Schemas:** Local instance 3306, Administration, Schemas.
- Query Editor:** Query 1, sakila-data*, SQL File 4*, Limit to 1000 rows.
- Result Grid:** Employee_Name (Peacock,Jane, Park,Margaret, Johnson,Steve) and Total_Sales (833.04, 775.40, 720.16).
- Action Output:** Shows the execution log with 34 entries, mostly successful (green checkmarks), indicating the execution of the query and its duration.
- Execution Plan:** Available on the right side of the interface.

11. Total sales \$ by media type

```

    select MediaType.Name as Media_Type, sum(invoiceLine.unitprice*invoiceLine.quantity) as Total_Sales from MediaType
    join Track on MediaType.MediaTypeId=Track.MediaTypeId
    join InvoiceLine on track.trackId=InvoiceLine.trackId
    group by Media_Type;
  
```

The screenshot shows the MySQL Workbench interface with the following details:

- Schemas:** Local instance 3306, Administration, Schemas.
- Query Editor:** Query 1, sakila-data*, SQL File 4*, Limit to 1000 rows.
- Result Grid:** Media_Type (Protected AAC audio file, MP3 audio file, Protected MPEG-4 video file, AAC audio file, Purchased AAC audio file) and Total_Sales (144.54, 1956.24, 220.89, 2.97, 3.96).
- Action Output:** Shows the execution log with 184 entries, mostly successful (green checkmarks), indicating the execution of the query and its duration.
- Execution Plan:** Available on the right side of the interface.

DESIGNING ADVANCED DATA ARCHITECTURE FOR BUSINESS INTELLIGENCE

ASSIGNMENT 2 – MySQL queries and outputs

00214365

12. Total sales \$ by genre

The screenshot shows the MySQL Workbench interface with a query editor and results grid. The query is:

```
50
51 • select Genre.Name as Genres, sum(invoiceLine.unitprice*invoiceLine.quantity) as Total_Sales from Genre
join Track on Genre.GenreId=Track.GenreId
join InvoiceLine on track.trackid=InvoiceLine.trackId
group by Genres order by Total_Sales desc;
```

The results grid displays the following data:

Genres	Total_Sales
Rock	826.05
Latin	382.14
Metal	261.36
Alternative & Punk	241.56
TV Shows	93.53
Jazz	78.40
Dance	63.59
Drama	57.71
R&B/Soul	40.59
Classical	40.59
Sci Fi & Fantasy	39.80
Reggae	29.70
Pop	27.72
Soundtrack	19.80
Comedy	17.91
Hip Hop/Rap	16.83
Bossa Nova	14.85
Alternative	13.86
World	12.57
Science Fiction	11.94
Heavy Metal	11.88
Electronic/Dance	11.88
Easy Listening	9.90
Rock And Roll	5.94

Query Completed

13. What are the total sales \$ by year

The screenshot shows the MySQL Workbench interface with a query editor and results grid. The query is:

```
48 • select year(InvoiceDate) as Years, sum(total) as Total_Sales from invoice group by Years;
```

A tooltip appears over the 'Rollback' button in the status bar, stating: "Rollback the current transaction. NOTE: all query tabs in the same connection share the same transaction. To have independent transactions, you must open a new connection."

The results grid displays the following data:

Years	Total_Sales
2009	440.48
2010	481.45
2011	469.58
2012	477.53
2013	450.58

Query Completed

DESIGNING ADVANCED DATA ARCHITECTURE FOR BUSINESS INTELLIGENCE

ASSIGNMENT 2 – MySQL queries and outputs

00214365

14. What are the total sales \$ by year-month

The screenshot shows the MySQL Workbench interface with a query editor and a result grid. The query is:

```

50
51 • select year(InvoiceDate) as Years, DATE_FORMAT(InvoiceDate, "%M") as Months, sum(total) as Total_Sales from invoice
52 group by Years, Months;
53
54
100% 1:53

```

The result grid displays the following data:

Years	Months	Total_Sales
2009	January	35.64
2009	February	37.62
2009	March	37.62
2009	April	37.62
2009	May	37.62
2009	June	37.62
2009	July	37.62
2009	August	37.62
2009	September	37.62
2009	October	37.62
2009	November	37.62
2009	December	37.62
2010	January	52.82
2010	February	46.62
2010	March	44.62
2010	April	37.62
2010	May	37.62
2010	June	37.62
2010	July	37.62
2010	August	37.62
2010	September	36.63
2010	October	37.62
2010	November	37.62
2010	December	37.62
2011	January	37.62
2011	February	37.62
2011	March	37.62
2011	April	51.82
2011	May	42.62
2011	June	50.62
2011	July	37.62
2011	August	37.62
2011	September	37.62
2011	October	37.62
2011	November	37.62
2011	December	37.62

Query Completed

15. What are the employees' name, birthday, hire date, years of working with company (assume as of 2013-12-31), address, city, state, country, title, manager and manager's title?

The screenshot shows the MySQL Workbench interface with a query editor and a result grid. The query is:

```

59
60 • select concat(employee.LastName, ' ', employee.FirstName) as Employee_Name, employee.title, date(employee.BirthDate) as BirthDate,
61 date(employee.HireDate) as HireDate, TIMESTAMPDIFF(YEAR, employee.HireDate, "2013-12-31") as Years_Of_Work,
62 ifNull(concat(manager.LastName, ' ', manager.FirstName), 'MANAGER THEMSELF') as Manager_Name, employee.title as Manager_Title, employee.address,
63 employee.city, employee.state, employee.country
64 from employee left join employee as manager on employee.employeeId=manager.ReportsTo;
65
66
100% 19:64

```

The result grid displays the following data:

Employee_Name	title	BirthDate	HireDate	Years_Of_Work	Manager_Name	Manager_Title	address	city	state	country
Adams, Andrew	General Manager	1962-02-18	2002-08-14	11	Edwards,Nancy	General Manager	11120 Jasper Ave NW	Edmonton	AB	Canada
Adams, Andrew	General Manager	1962-02-18	2002-08-14	11	Mitchell,Michael	General Manager	11120 Jasper Ave NW	Edmonton	AB	Canada
Edwards,Nancy	Sales Manager	1958-12-08	2002-05-01	11	Peacock,Jane	Sales Manager	825 8 Ave SW	Calgary	AB	Canada
Edwards,Nancy	Sales Manager	1958-12-08	2002-05-01	11	Park,Margaret	Sales Manager	825 8 Ave SW	Calgary	AB	Canada
Edwards,Nancy	Sales Manager	1958-12-08	2002-05-01	11	Johnson,Steve	Sales Manager	825 8 Ave SW	Calgary	AB	Canada
Parsons,Mark	Sales Support Agent	1970-04-01	2003-05-01	11	MANAGER THEMSELF	Sales Support Agent	1120 17 Street SW	Calgary	AB	Canada
Parsons,Mark	Sales Support Agent	1970-04-01	2003-05-01	11	MANAGER THEMSELF	Sales Support Agent	680 10 Street SW	Calgary	AB	Canada
King,Robert	IT Staff	1970-05-29	2004-01-02	9	MANAGER THEMSELF	IT Staff	727B 41 Ave	Calgary	AB	Canada
Callahan,Laura	IT Staff	1968-01-09	2004-03-04	9	MANAGER THEMSELF	IT Staff	923 7 ST NW	Lethbridge	AB	Canada

Result 144

Action Output: c

Time	Action	Response	Duration / Fetch Time
173 20:38:35	select concat(employee.LastName, ' ', employee.FirstName) as Employee_Name, employee.title, date(employee.Birth... 12 row(s) returned		0.0020 sec / 0.00003...

Query Completed

DESIGNING ADVANCED DATA ARCHITECTURE FOR BUSINESS INTELLIGENCE

ASSIGNMENT 2 – MySQL queries and outputs

00214365

16. What are the total sales \$ by employee age at the time of the invoice date?

```

    select concat(employee.LastName, ' ', employee.FirstName) as Employee_Name,
    TIMESTAMPDIFF(YEAR, employee.BirthDate, invoice.InvoiceDate) as Age_At_InvoiceDate, sum(total) as Total_Sales from employee
    join customer on employee.employeeId=customer.supportRepId
    join invoice on customer.customerId = invoice.customerId
    group by Employee_Name, Age_At_InvoiceDate;
  
```

Employee_Name	Age_At_InvoiceDate	Total_Sales
Peacock,Jane	36	169.65
Peacock,Jane	37	199.19
Peacock,Jane	39	166.40
Peacock,Jane	38	143.55
Peacock,Jane	40	55.45
Peacock,Jane	35	108.90
Park,Margaret	61	109.89
Park,Margaret	62	141.59
Park,Margaret	63	127.75
Park,Margaret	64	178.33
Park,Margaret	66	40.59
Park,Margaret	65	177.27
Johnson,Steve	43	24.75
Johnson,Steve	44	197.16
Johnson,Steve	46	144.62
Johnson,Steve	47	121.85
Johnson,Steve	45	119.85
Johnson,Steve	48	117.93

Result 149 | Read Only

Action Output | Time Action | Response | Duration / Fetch Time

178 21:16:01 select concat(employee.LastName, ' ', employee.FirstName) as Employee_Name, TIMESTAMPDIFF(YEAR, employee... 18 row(s) returned 0.0034 sec / 0.00001...

Query Completed

17. What are the total sales \$ by employees who are in their 30s, 40s, 50s and 60s (employee age at the time of the invoice date)

```

    select concat(employee.LastName, ' ', employee.FirstName) as Employee_Name,
    sum(case when (TIMESTAMPDIFF(YEAR, employee.BirthDate, invoice.InvoiceDate)>=30 and
    TIMESTAMPDIFF(YEAR, employee.BirthDate, invoice.InvoiceDate)<=39) then invoice.total else 'N/A' end) as 'Sales_at_30s',
    sum(case when (TIMESTAMPDIFF(YEAR, employee.BirthDate, invoice.InvoiceDate)>=40 and
    TIMESTAMPDIFF(YEAR, employee.BirthDate, invoice.InvoiceDate)<=49) then invoice.total else 'N/A' end) as 'Sales_at_40s',
    sum(case when (TIMESTAMPDIFF(YEAR, employee.BirthDate, invoice.InvoiceDate)>=50 and
    TIMESTAMPDIFF(YEAR, employee.BirthDate, invoice.InvoiceDate)<=59) then invoice.total else 'N/A' end) as 'Sales_at_50s',
    sum(case when (TIMESTAMPDIFF(YEAR, employee.BirthDate, invoice.InvoiceDate)>=60 and
    TIMESTAMPDIFF(YEAR, employee.BirthDate, invoice.InvoiceDate)<=69) then invoice.total else 'N/A' end) as 'Sales_at_60s'
  
```

Employee_Name	Sales_at_30s	Sales_at_40s	Sales_at_50s	Sales_at_60s
Peacock,Jane	777.590000000012	55.45	0	0
Park,Margaret	0	0	775.400000000011	0
Johnson,Steve	0	720.160000000001	0	0

Result 173 | Read Only

Action Output | Time Action | Response | Duration / Fetch Time

203 22:10:59 select concat(employee.LastName, ' ', employee.FirstName) as Employee_Name, max(case when (TIMESTAMPDIFF(... 3 row(s) returned 0.0013 sec / 0.00001...

204 22:19:57 select concat(employee.LastName, ' ', employee.FirstName) as Employee_Name, sum(case when (TIMESTAMPDIFF(... 3 row(s) returned 0.0031 sec / 0.00001...

205 22:20:47 select concat(employee.LastName, ' ', employee.FirstName) as Employee_Name, sum(case when (TIMESTAMPDIFF(... 3 row(s) returned 0.0070 sec / 0.00001...

206 22:21:51 select concat(employee.LastName, ' ', employee.FirstName) as Employee_Name, sum(case when (TIMESTAMPDIFF(... 3 row(s) returned 0.0056 sec / 0.00001...

Query Completed