Fall
2023

MGMT 58200: SQL Project for Market Data Forecast

GROUP_3

PRATHYUSHA REDDY MIDUDHULA
VENKATA SAI TEJA GANGUMALLA
GOUTHAM KUMAR VEMASANI
HARISH DATTA CHITNENI
MAX MCTIGUE
SANJAY KATYAL

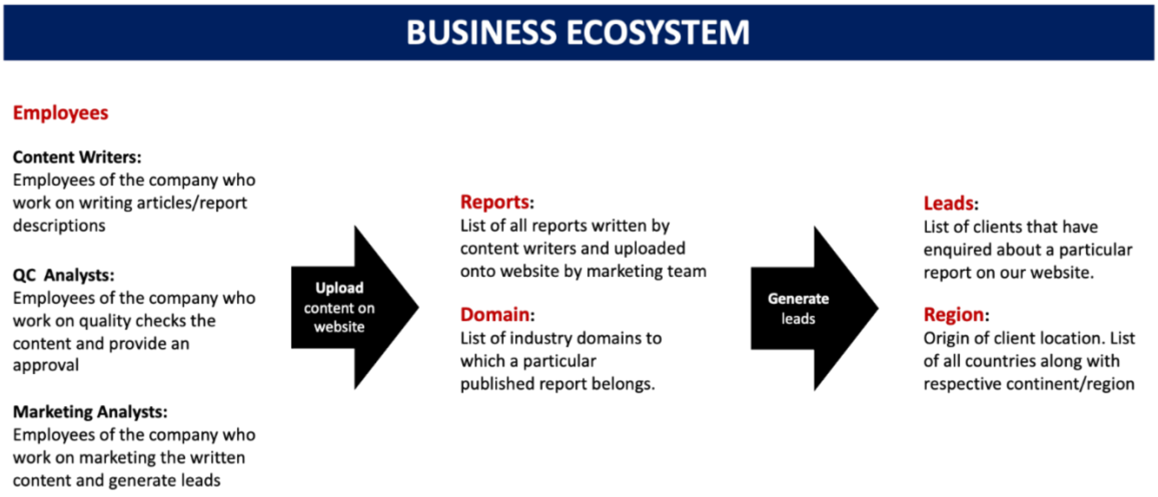
PURDUE UNIVERSITY | West Lafayette

Table of Contents

About the Company:

- Market Data Forecast (MDF) is a provider of syndicated and custom-made market research, business intelligence and consulting services on gamut of sectors across the globe.
- Primarily, the company offers pre-published market research reports for clients. In addition to this, the company only offers custom market/business research services in-line with client objectives.
- The company has been associated with many Fortune500 listed companies and offers its services spanning across 12+ industry domains and 60+ sub-categories.

Business Ecosystem and Process:



BUSINESS ECOSYSTEM

Employees

Content Writers:
Employees of the company who work on writing articles/report descriptions

QC Analysts:
Employees of the company who work on quality checks the content and provide an approval

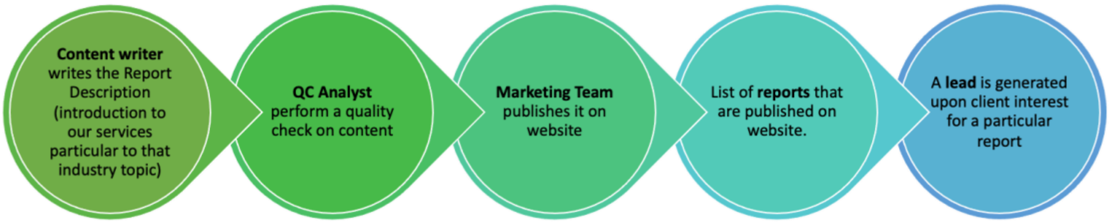
Marketing Analysts:
Employees of the company who work on marketing the written content and generate leads

Reports:
List of all reports written by content writers and uploaded onto website by marketing team

Domain:
List of industry domains to which a particular published report belongs.

Leads:
List of clients that have enquired about a particular report on our website.

Region:
Origin of client location. List of all countries along with respective continent/region



Content writer writes the Report Description (introduction to our services particular to that industry topic)

QC Analyst perform a quality check on content

Marketing Team publishes it on website

List of **reports** that are published on website.

A **lead** is generated upon client interest for a particular report

Business Case objectives:

- → Migrate the existing Excel-based data system to a robust SQL database to enhance data integrity, security, and accessibility.
- → Analyze the total volume of content created by the company to assess its impact on overall company objectives.
- → Develop an SQL process to calculate the monthly payment for content creators based on a per-word payment model.
- → Implement an SQL-based performance evaluation system to assess the quality and effectiveness of content created by individual content writers. Provide recommendations for improvements.
- → Evaluate the performance of each marketing team member using SQL-based metrics and KPIs to determine their contributions to the company's marketing efforts.
- → Monthly Performance Growth Tracking
- → Establish SQL-based tracking and reporting mechanisms to monitor the month-over-month growth rate in performance metrics for both content creation and marketing activities.

Attributes Description:

Entity: Employees

EmployeeID: Unique identifier for each employee.

Name: The name of the employee.

Title: The job title or position of the employee.

Pricer_Per_Word: The pricing rate per word for the employee's services.

Commision_rate: The commission rate for the employee.

Commision_limit: The maximum commission limit for the employee.

Entity: Reports

ReportID: Unique identifier for each report.

AddedDate: Date when the report title was added by marketing team member.

WordCount: Minimum number of words of content required for the report.

ContentWriterID: Identifier of the content writer who authored the report.

WrittenDate: Date when the report was written by content writer.

QC_Date: Date when the report underwent quality control (QC).

QCManagerID: Identifier of the QC manager responsible for QC.

QC_Status: Status of the quality control process.

UpdatedDate: Date when the report was last updated.

Entity: Domain

DomainID: Unique identifier for each domain.

Domain: The name or description of the domain.

Entity: Region

Country: Name of a country.

Continent: The continent to which the country belongs.

Entity: Leads

id: Unique identifier for each lead.

report_id: Identifier linking the lead to a specific report.

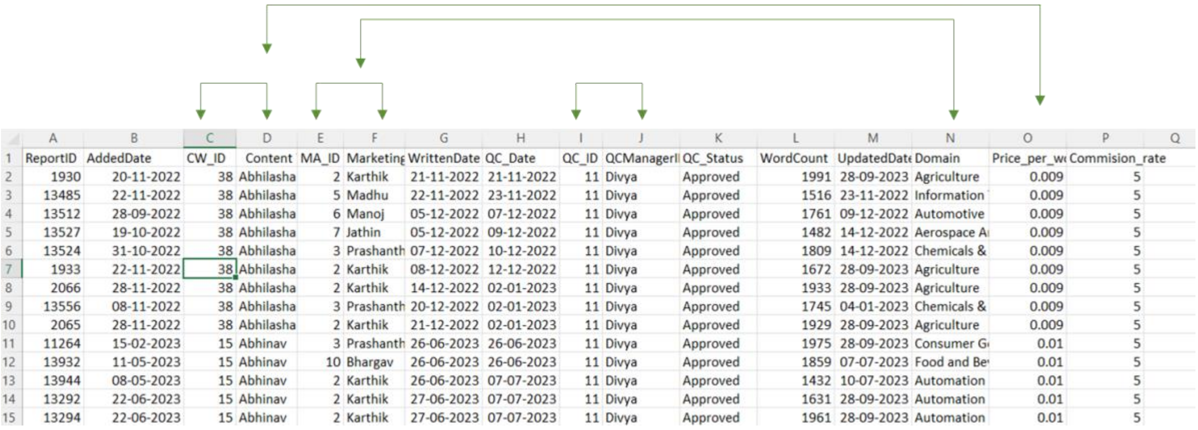
created_at: Date and time when the lead was created.

country: The country associated with the lead.

Normalization Analysis:

Market Data Forecast has provided the data in three tables, first is the Report_Table and second is the Leads_Table and the third is employees table, which are all in two normal form:

- There are no multi-valued or composite attributes
- They **do not have any partial dependencies**, i.e. all data is related to one primary key. “ReportID” in Report_Table and “id” in Leads_Table



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	ReportID	AddedDate	CW_ID	Content	MA_ID	Marketing	WrittenDate	QC_Date	QC_ID	QCManagerI	QC_Status	WordCount	UpdatedDate	Domain	Price_per_w	Commision_rate	
2	1930	20-11-2022	38	Abhilasha	2	Karthik	21-11-2022	21-11-2022	11	Divya	Approved	1991	28-09-2023	Agriculture	0.009	5	
3	13485	22-11-2022	38	Abhilasha	5	Madhu	22-11-2022	23-11-2022	11	Divya	Approved	1516	23-11-2022	Information	0.009	5	
4	13512	28-09-2022	38	Abhilasha	6	Manoj	05-12-2022	07-12-2022	11	Divya	Approved	1761	09-12-2022	Automotive	0.009	5	
5	13527	19-10-2022	38	Abhilasha	7	Jathin	05-12-2022	09-12-2022	11	Divya	Approved	1482	14-12-2022	Aerospace A	0.009	5	
6	13524	31-10-2022	38	Abhilasha	3	Prashanth	07-12-2022	10-12-2022	11	Divya	Approved	1809	14-12-2022	Chemicals &	0.009	5	
7	1933	22-11-2022	38	Abhilasha	2	Karthik	08-12-2022	12-12-2022	11	Divya	Approved	1672	28-09-2023	Agriculture	0.009	5	
8	2066	28-11-2022	38	Abhilasha	2	Karthik	14-12-2022	02-01-2023	11	Divya	Approved	1933	28-09-2023	Agriculture	0.009	5	
9	13556	08-11-2022	38	Abhilasha	3	Prashanth	20-12-2022	02-01-2023	11	Divya	Approved	1745	04-01-2023	Chemicals &	0.009	5	
10	2065	28-11-2022	38	Abhilasha	2	Karthik	21-12-2022	02-01-2023	11	Divya	Approved	1929	28-09-2023	Agriculture	0.009	5	
11	11264	15-02-2023	15	Abhinav	3	Prashanth	26-06-2023	26-06-2023	11	Divya	Approved	1975	28-09-2023	Consumer G	0.01	5	
12	13932	11-05-2023	15	Abhinav	10	Bhargav	26-06-2023	26-06-2023	11	Divya	Approved	1859	07-07-2023	Food and Be	0.01	5	
13	13944	08-05-2023	15	Abhinav	2	Karthik	26-06-2023	07-07-2023	11	Divya	Approved	1432	10-07-2023	Automation	0.01	5	
14	13292	22-06-2023	15	Abhinav	2	Karthik	27-06-2023	07-07-2023	11	Divya	Approved	1631	28-09-2023	Automation	0.01	5	
15	13294	22-06-2023	15	Abhinav	2	Karthik	27-06-2023	07-07-2023	11	Divya	Approved	1961	28-09-2023	Automation	0.01	5	

Converting the 2 Normal Form to 3 Normal Form:

There are transitive dependencies, in both the tables, Report_Table and the Leads_Table.

In **Reports_Table**:

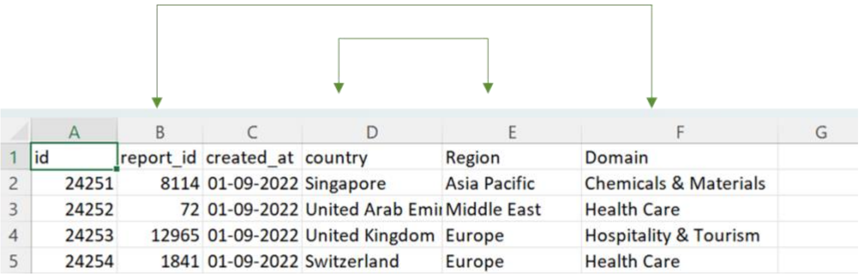
ReportID → DomainID → Domain.

ReportID → ContentWriterID → ContentWriter.

ReportID → QCManagerID → QCManager.

ReportID → ContentWriterID → price_per_word.

ReportID → DomainID → MarketingAnalyst.



	A	B	C	D	E	F	G
1	id	report_id	created_at	country	Region	Domain	
2	24251	8114	01-09-2022	Singapore	Asia Pacific	Chemicals & Materials	
3	24252	72	01-09-2022	United Arab Emi	Middle East	Health Care	
4	24253	12965	01-09-2022	United Kingdom	Europe	Hospitality & Tourism	
5	24254	1841	01-09-2022	Switzerland	Europe	Health Care	

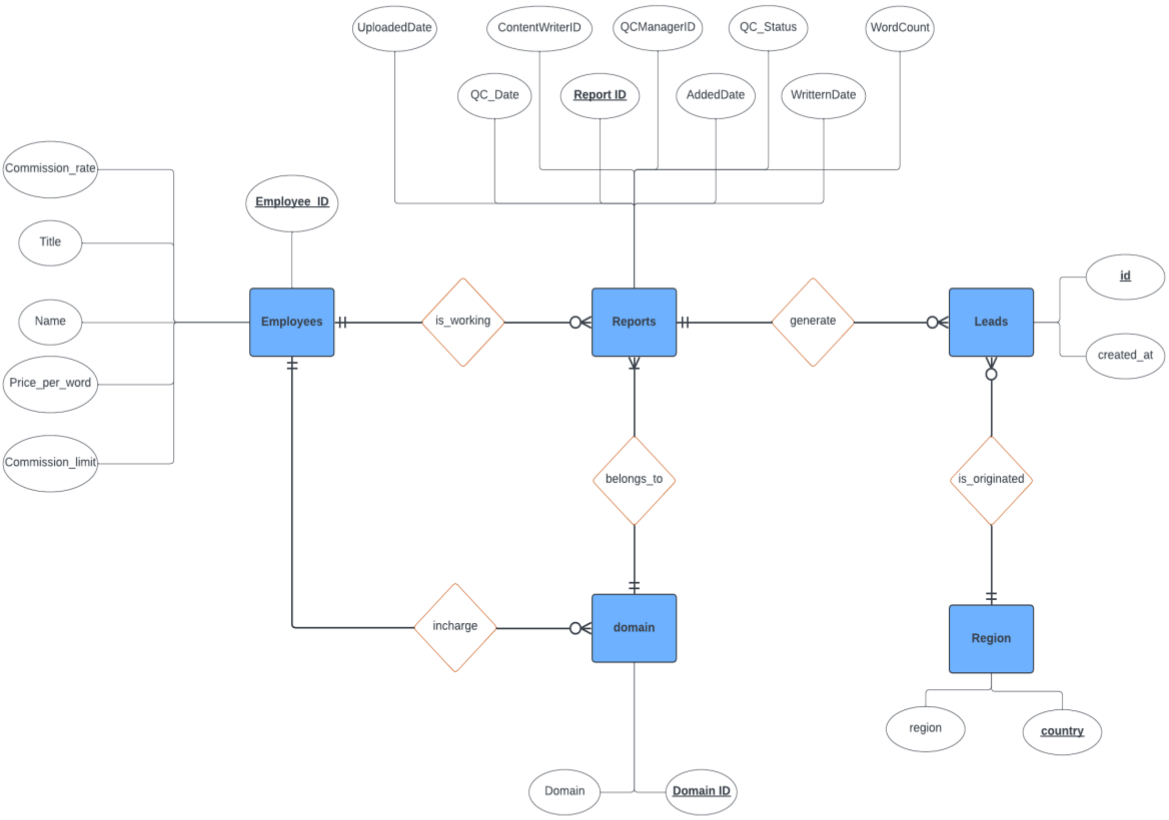
In **Leads_Table**:

Id → country → continent

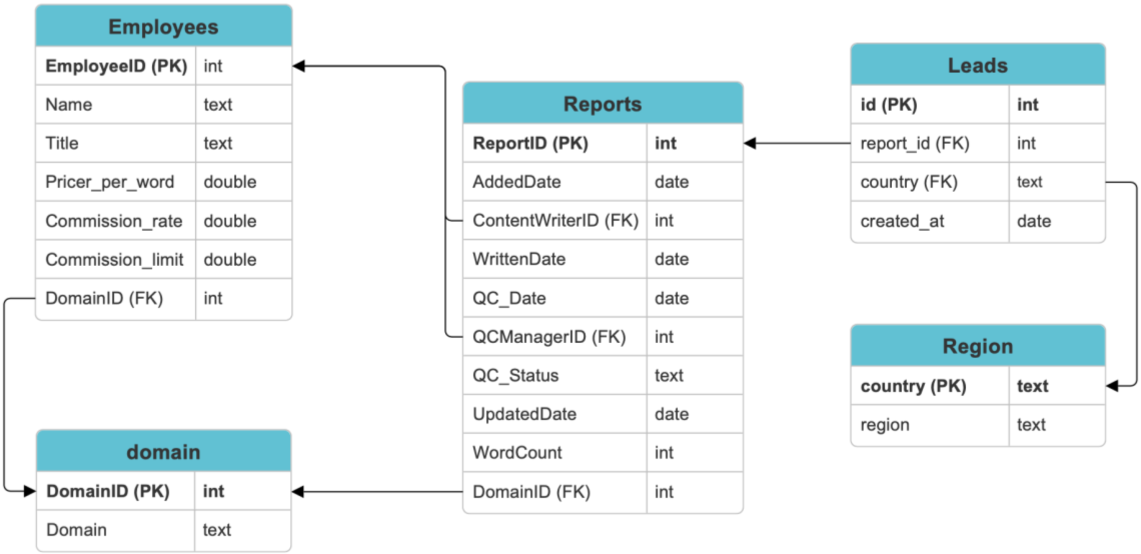
Id → ReportID → Domain

To remove these transitive dependencies, we moved few attributes to new tables.

ER Diagram:



ER Schema:



Employees	
EmployeeID (PK)	int
Name	text
Title	text
Pricer_per_word	double
Commission_rate	double
Commission_limit	double
DomainID (FK)	int

Reports	
ReportID (PK)	**int**
AddedDate	date
ContentWriterID (FK)	int
WrittenDate	date
QC_Date	date
QCManagerID (FK)	int
QC_Status	text
UpdatedDate	date
WordCount	int
DomainID (FK)	int

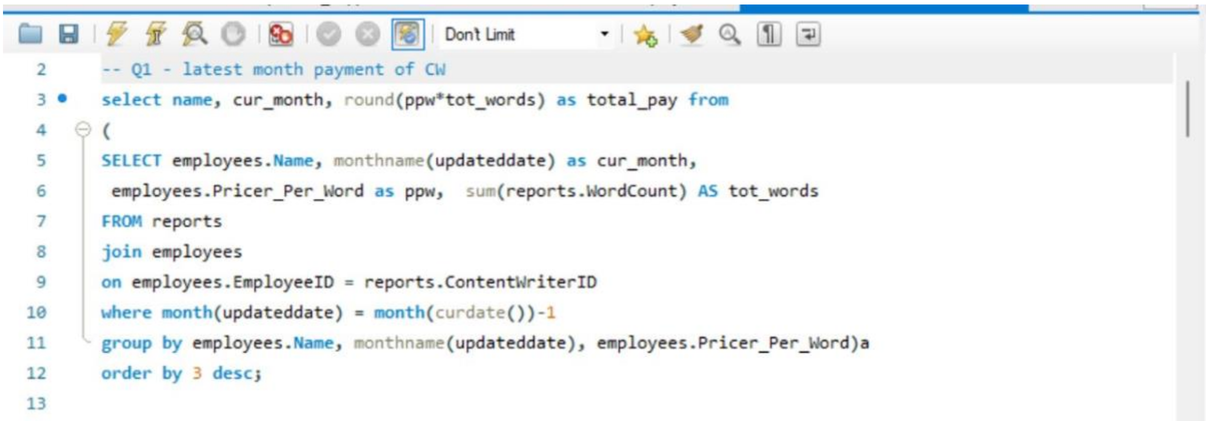
Leads	
id (PK)	**int**
report_id (FK)	int
country (FK)	text
created_at	date

Region	
country (PK)	**text**
region	text

domain	
DomainID (PK)	**int**
Domain	text

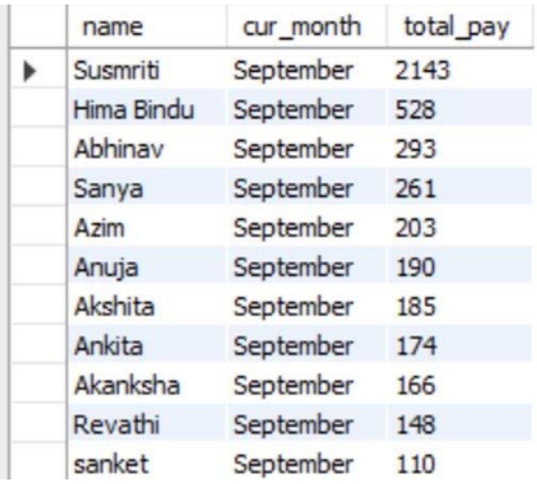
SQL QUERIES AND OUTPUT

Objective 1: Find the amount of money to be paid to each Content writer for the current month till date.



```
-- Q1 - latest month payment of CW
select name, cur_month, round(ppw*tot_words) as total_pay from
(
SELECT employees.Name, monthname(updateddate) as cur_month,
 employees.Pricer_Per_Word as ppw,  sum(reports.WordCount) AS tot_words
FROM reports
join employees
on employees.EmployeeID = reports.ContentWriterID
where month(updateddate) = month(curdate())-1
group by employees.Name, monthname(updateddate), employees.Pricer_Per_Word)a
order by 3 desc;
```

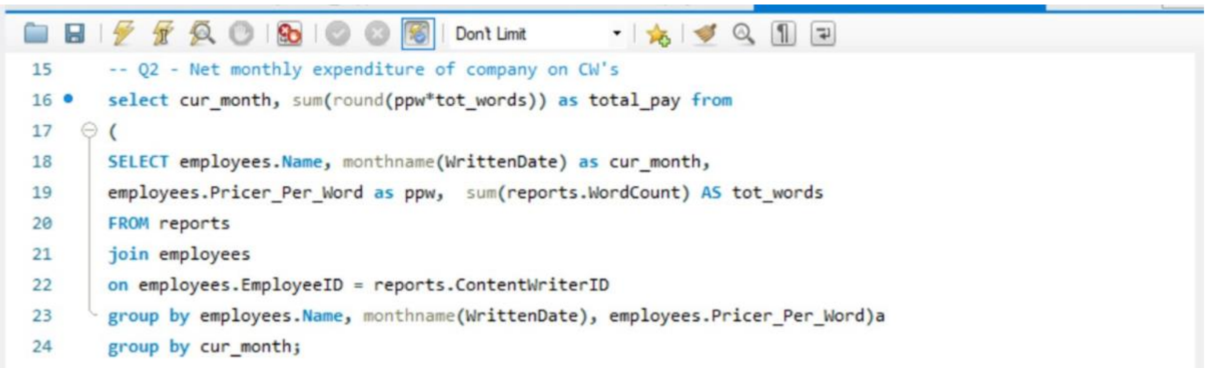
Output:



name	cur_month	total_pay
Susmriti	September	2143
Hima Bindu	September	528
Abhinav	September	293
Sanya	September	261
Azim	September	203
Anuja	September	190
Akshita	September	185
Ankita	September	174
Akanksha	September	166
Revathi	September	148
sanket	September	110

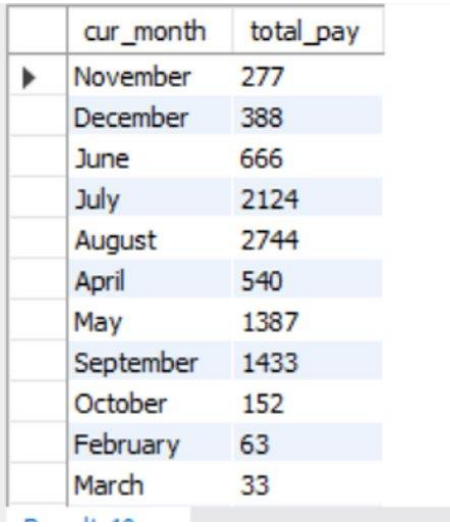
Business Application: This gives us the amount we need to pay each of our employees for the current month and this can help us in keeping track of payments to be made.

Objective 2 - Total amount to be spent by the company in the current month.



```
-- Q2 - Net monthly expenditure of company on CW's
select cur_month, sum(round(ppw*tot_words)) as total_pay from
(
SELECT employees.Name, monthname(WrittenDate) as cur_month,
employees.Pricer_Per_Word as ppw,  sum(reports.WordCount) AS tot_words
FROM reports
join employees
on employees.EmployeeID = reports.ContentWriterID
group by employees.Name, monthname(WrittenDate), employees.Pricer_Per_Word)a
group by cur_month;
```

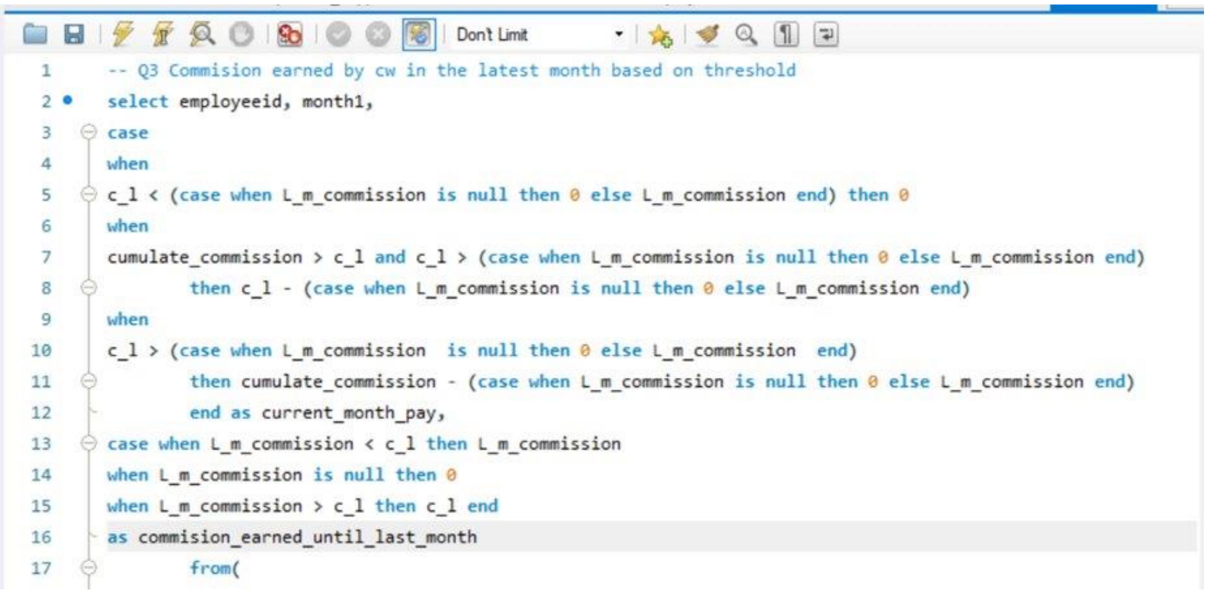
Output:



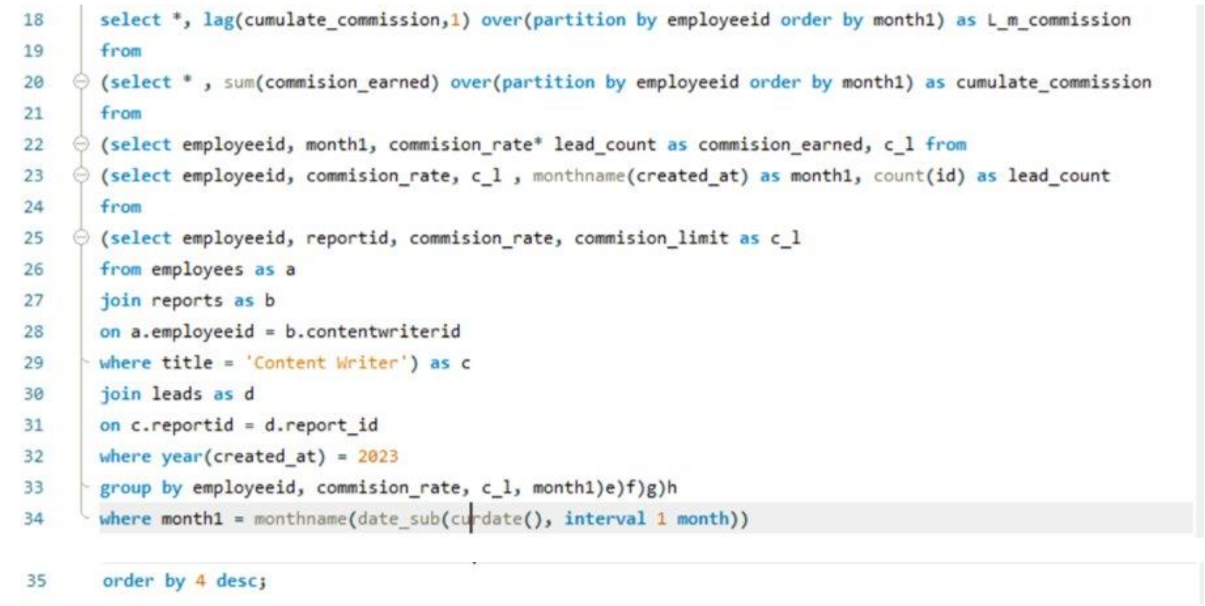
cur_month	total_pay
November	277
December	388
June	666
July	2124
August	2744
April	540
May	1387
September	1433
October	152
February	63
March	33

Business Application: This gives us the amount we need to pay in total to all our content writers in the current month and this can help in gauging how much budget we need for the month.

Objective 3 - Commission earned by Content writers in the latest month based on threshold.



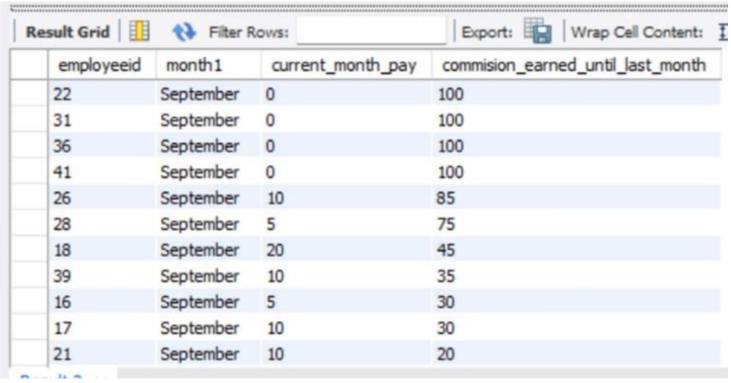
```
-- Q3 Commision earned by cw in the latest month based on threshold
select employeeid, month1,
case
when
c_l < (case when L_m_commission is null then 0 else L_m_commission end) then 0
when
cumulate_commission > c_l and c_l > (case when L_m_commission is null then 0 else L_m_commission end)
        then c_l - (case when L_m_commission is null then 0 else L_m_commission end)
when
c_l > (case when L_m_commission  is null then 0 else L_m_commission  end)
        then cumulate_commission - (case when L_m_commission is null then 0 else L_m_commission end)
        end as current_month_pay,
case when L_m_commission < c_l then L_m_commission
when L_m_commission is null then 0
when L_m_commission > c_l then c_l end
as commision_earned_until_last_month
        from(
```



```
select *, lag(cumulate_commission,1) over(partition by employeeid order by month1) as L_m_commission
from
(select * , sum(commision_earned) over(partition by employeeid order by month1) as cumulate_commission
from
(select employeeid, month1, commision_rate* lead_count as commision_earned, c_l from
(select employeeid, commision_rate, c_l , monthname(created_at) as month1, count(id) as lead_count
from
(select employeeid, reportid, commision_rate, commision_limit as c_l
from employees as a
join reports as b
on a.employeeid = b.contentwriterid
where title = 'Content Writer') as c
join leads as d
on c.reportid = d.report_id
where year(created_at) = 2023
group by employeeid, commision_rate, c_l, month1)e)f)g)h
where month1 = monthname(date_sub(curdate(), interval 1 month))

order by 4 desc;
```

Output:

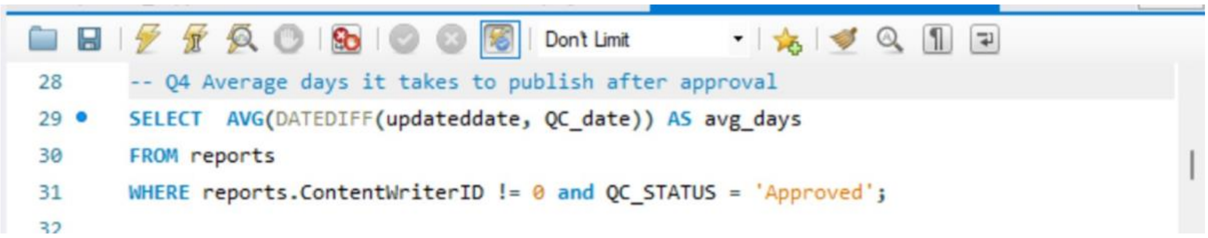


Result Grid | Filter Rows: | Export: | Wrap Cell Content:

employeeid	month1	current_month_pay	commision_earned_until_last_month
22	September	0	100
31	September	0	100
36	September	0	100
41	September	0	100
26	September	10	85
28	September	5	75
18	September	20	45
39	September	10	35
16	September	5	30
17	September	10	30
21	September	10	20

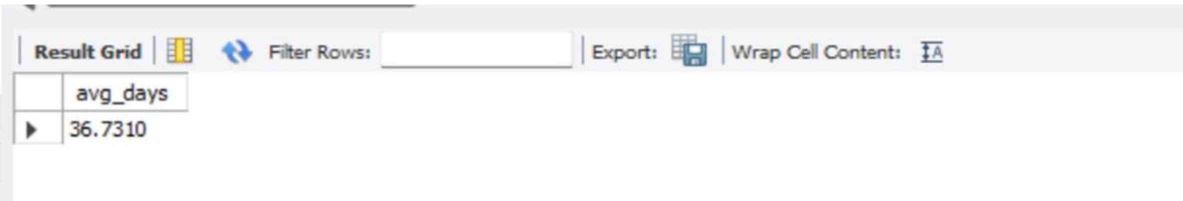
Business Application: This gives us the amount of commission earned by our content writers, based on the performance of their articles. This helps in understanding our bonus payment budget

Objective 4 – Calculating the average number of days it takes to Publish an article after it has passed the Quality check



```sql
-- Q4 Average days it takes to publish after approval
SELECT  AVG(DATEDIFF(updateddate, QC_date)) AS avg_days
FROM reports
WHERE reports.ContentWriterID != 0 and QC_STATUS = 'Approved';
```

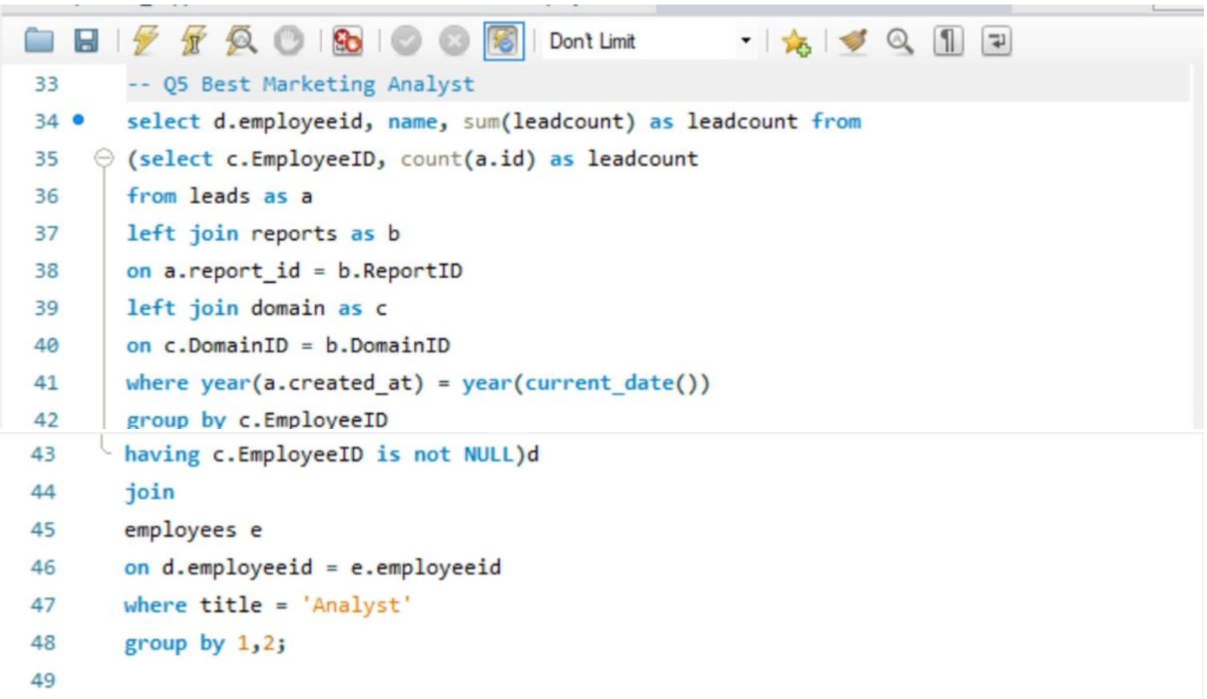
Output:



```
  avg_days
▶ 36.7310
```

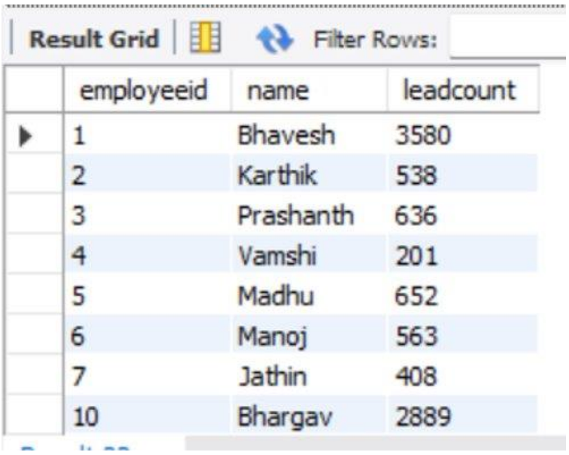
Business Application: Helps in understanding the amount of time it takes to complete the Quality check process, to help make it more efficient if necessary.

Objective 5- Arranging the Marketing analysts by the lead count their report has produced



```sql
-- Q5 Best Marketing Analyst
select d.employeeid, name, sum(leadcount) as leadcount from
(select c.EmployeeID, count(a.id) as leadcount
from leads as a
left join reports as b
on a.report_id = b.ReportID
left join domain as c
on c.DomainID = b.DomainID
where year(a.created_at) = year(current_date())
group by c.EmployeeID
having c.EmployeeID is not NULL)d
join
employees e
on d.employeeid = e.employeeid
where title = 'Analyst'
group by 1,2;
```

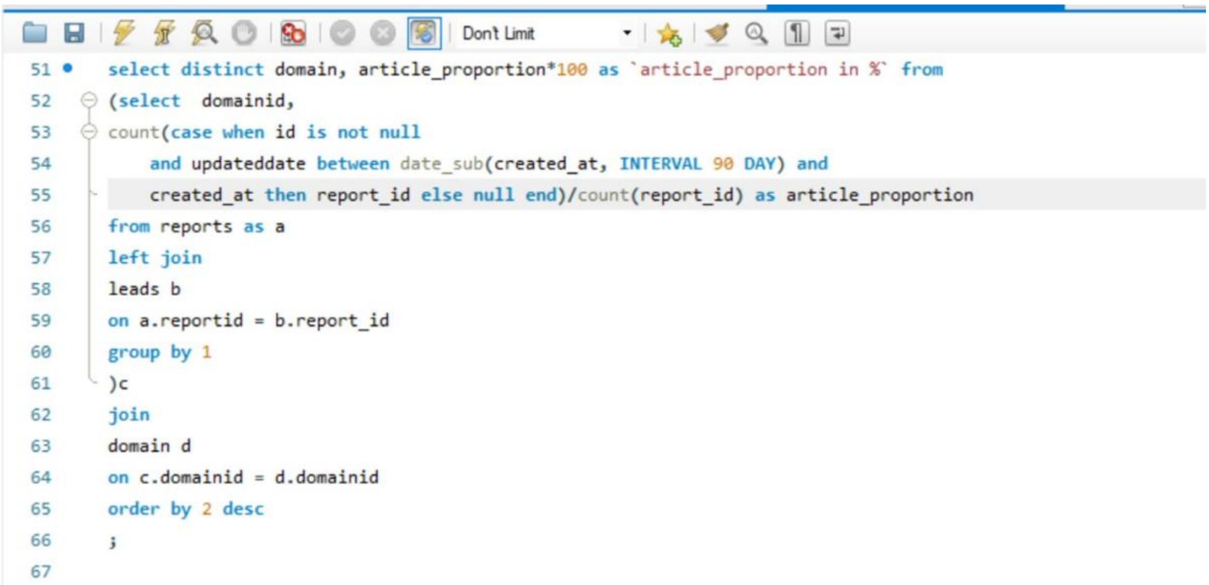
Output:



employeeid	name	leadcount
1	Bhavesh	3580
2	Karthik	538
3	Prashanth	636
4	Vamshi	201
5	Madhu	652
6	Manoj	563
7	Jathin	408
10	Bhargav	2889

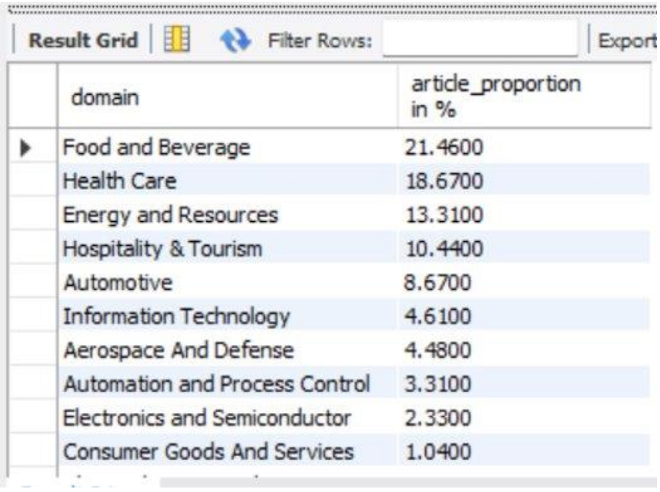
Business Application: To figure out the top performing marketing analysts who have produced the most lead count on their article, so as to know the top performers.

Objective 6 – What is the proportion of reports in each domain that are generating leads within the first 3 months.



```
select distinct domain, article_proportion*100 as `article_proportion in %` from
(select  domainid,
count(case when id is not null
    and updateddate between date_sub(created_at, INTERVAL 90 DAY) and
    created_at then report_id else null end)/count(report_id) as article_proportion
from reports as a
left join
leads b
on a.reportid = b.report_id
group by 1
)c
join
domain d
on c.domainid = d.domainid
order by 2 desc
;
```

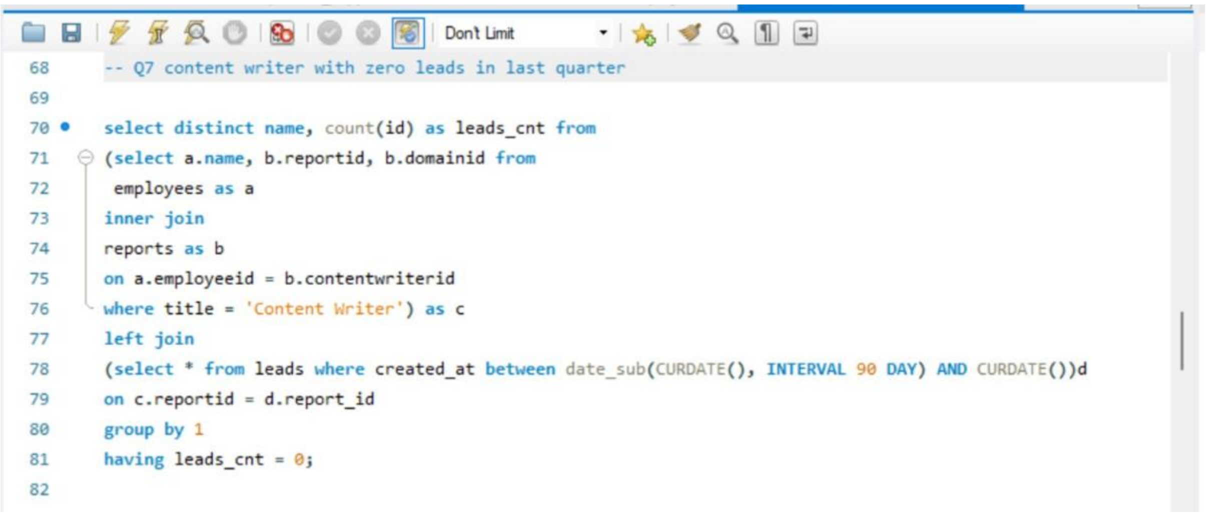
Output:



domain	article_proportion in %
Food and Beverage	21.4600
Health Care	18.6700
Energy and Resources	13.3100
Hospitality & Tourism	10.4400
Automotive	8.6700
Information Technology	4.6100
Aerospace And Defense	4.4800
Automation and Process Control	3.3100
Electronics and Semiconductor	2.3300
Consumer Goods And Services	1.0400

Business Application: From this data, we can see what proportion of the articles from each domain are able to generate leads within the first three months, this gives us an idea about the best performing domains.

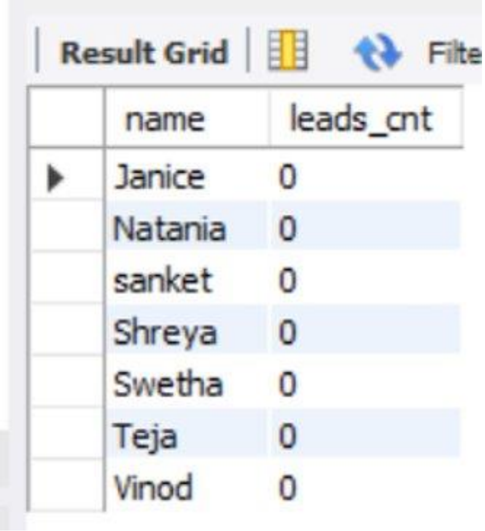
Objective 7– To find all the content writers with zero leads in last quarter



```
-- Q7 content writer with zero leads in last quarter

select distinct name, count(id) as leads_cnt from
(select a.name, b.reportid, b.domainid from
 employees as a
inner join
reports as b
on a.employeeid = b.contentwriterid
where title = 'Content Writer') as c
left join
(select * from leads where created_at between date_sub(CURDATE(), INTERVAL 90 DAY) AND CURDATE())d
on c.reportid = d.report_id
group by 1
having leads_cnt = 0;
```

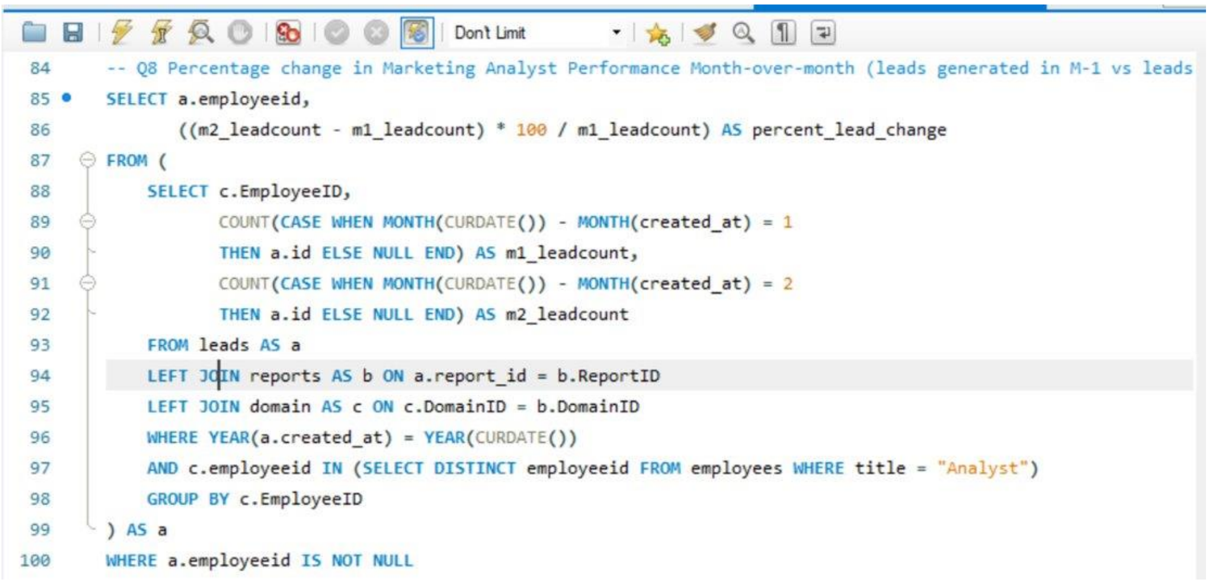
Output:



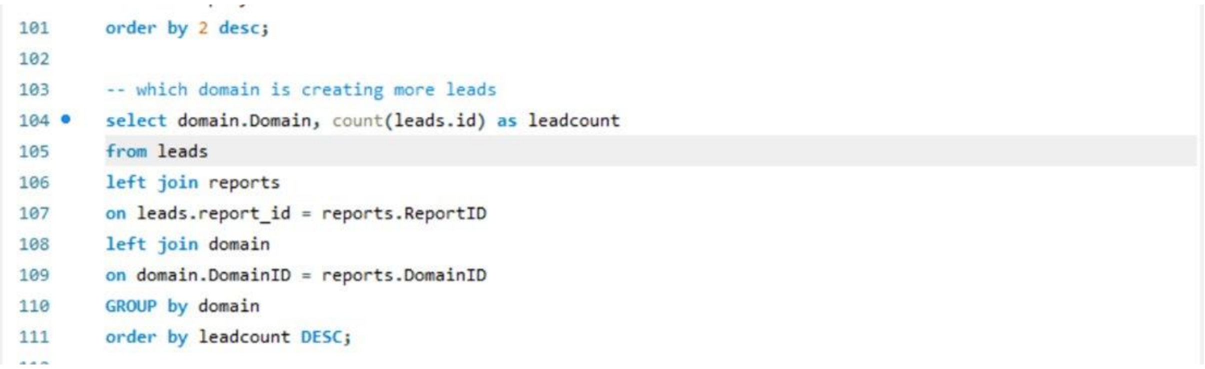
name	leads_cnt
Janice	0
Natania	0
sanket	0
Shreya	0
Swetha	0
Teja	0
Vinod	0

Business Application: From this list we can see the lowest performing content writers for the last four months. It can help deciding the future pay per word for the writers based on performance.

Objective 8 - Percentage change in Marketing Analyst Performance Month-over-month (leads generated in M-1 vs leads generated in M-2)



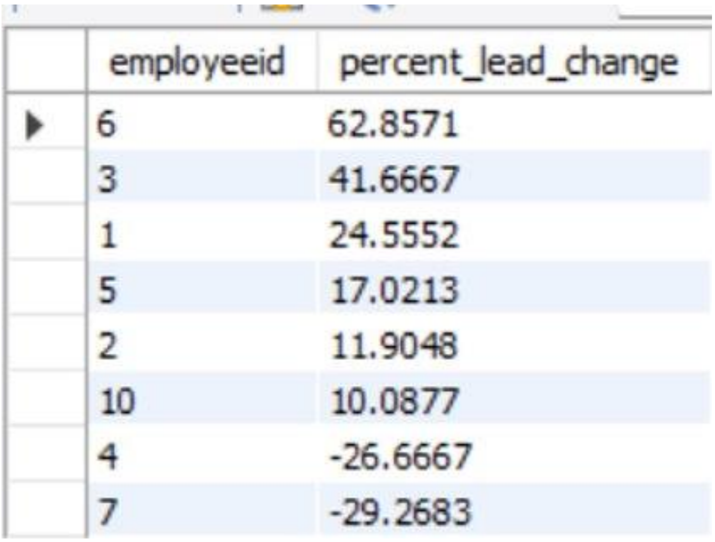
```
-- Q8 Percentage change in Marketing Analyst Performance Month-over-month (leads generated in M-1 vs leads
SELECT a.employeeid,
       ((m2_leadcount - m1_leadcount) * 100 / m1_leadcount) AS percent_lead_change
FROM (
    SELECT c.EmployeeID,
           COUNT(CASE WHEN MONTH(CURDATE()) - MONTH(created_at) = 1
           THEN a.id ELSE NULL END) AS m1_leadcount,
           COUNT(CASE WHEN MONTH(CURDATE()) - MONTH(created_at) = 2
           THEN a.id ELSE NULL END) AS m2_leadcount
    FROM leads AS a
    LEFT JOIN reports AS b ON a.report_id = b.ReportID
    LEFT JOIN domain AS c ON c.DomainID = b.DomainID
    WHERE YEAR(a.created_at) = YEAR(CURDATE())
    AND c.employeeid IN (SELECT DISTINCT employeeid FROM employees WHERE title = "Analyst")
    GROUP BY c.EmployeeID
) AS a
WHERE a.employeeid IS NOT NULL
```



```
order by 2 desc;

-- which domain is creating more leads
select domain.Domain, count(leads.id) as leadcount
from leads
left join reports
on leads.report_id = reports.ReportID
left join domain
on domain.DomainID = reports.DomainID
GROUP by domain
order by leadcount DESC;
```

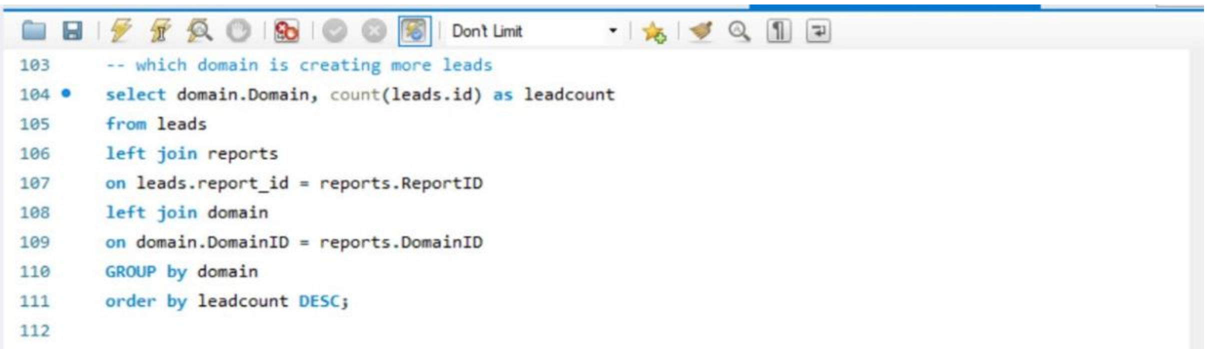
Output:



	employeeid	percent_lead_change
▶	6	62.8571
	3	41.6667
	1	24.5552
	5	17.0213
	2	11.9048
	10	10.0877
	4	-26.6667
	7	-29.2683

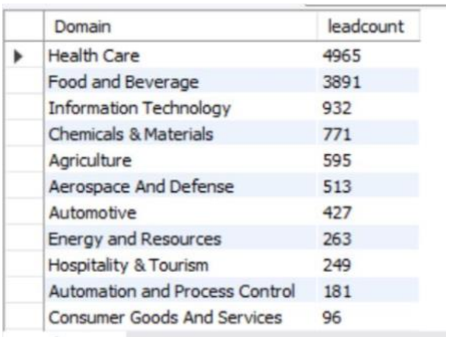
Business Application: We can use this as a metric to see how performances are changing month over month, based on the leads that each marketing analyst is generating in a given month.

Objective 9 – To find the most lead generating Domain



```
-- which domain is creating more leads
select domain.Domain, count(leads.id) as leadcount
from leads
left join reports
on leads.report_id = reports.ReportID
left join domain
on domain.DomainID = reports.DomainID
GROUP by domain
order by leadcount DESC;
```

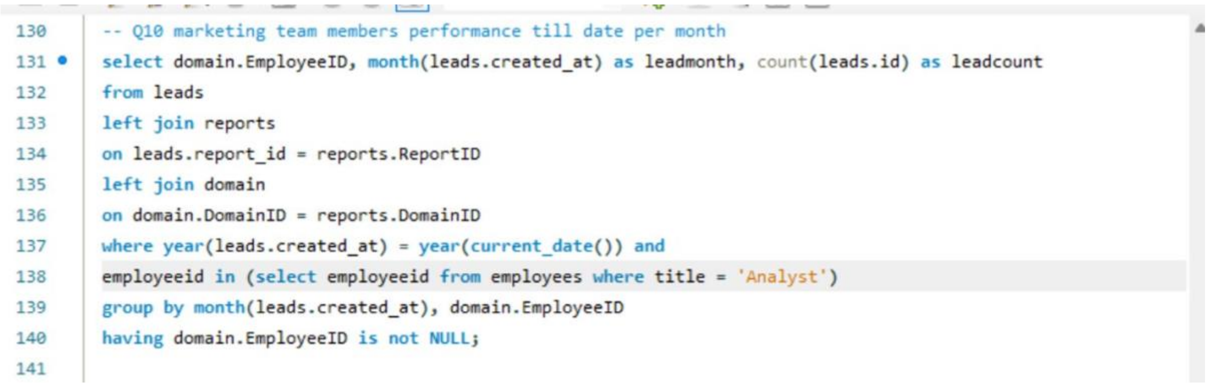
Output:



Domain	leadcount
Health Care	4965
Food and Beverage	3891
Information Technology	932
Chemicals & Materials	771
Agriculture	595
Aerospace And Defense	513
Automotive	427
Energy and Resources	263
Hospitality & Tourism	249
Automation and Process Control	181
Consumer Goods And Services	96

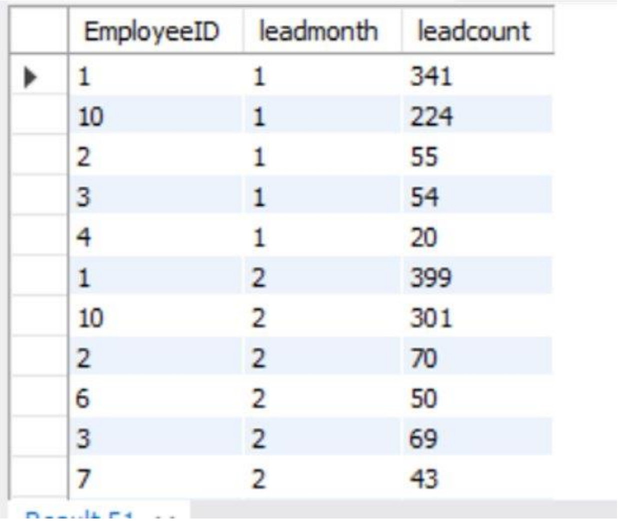
Business Application: From this table we can see which domain/industry related reports are generating the most leads and thus we can allot more resources to the top performing domains.

Objective 10 - Find the performance of marketing team members monthly.



```
-- Q10 marketing team members performance till date per month
select domain.EmployeeID, month(leads.created_at) as leadmonth, count(leads.id) as leadcount
from leads
left join reports
on leads.report_id = reports.ReportID
left join domain
on domain.DomainID = reports.DomainID
where year(leads.created_at) = year(current_date()) and
employeeid in (select employeeid from employees where title = 'Analyst')
group by month(leads.created_at), domain.EmployeeID
having domain.EmployeeID is not NULL;
```

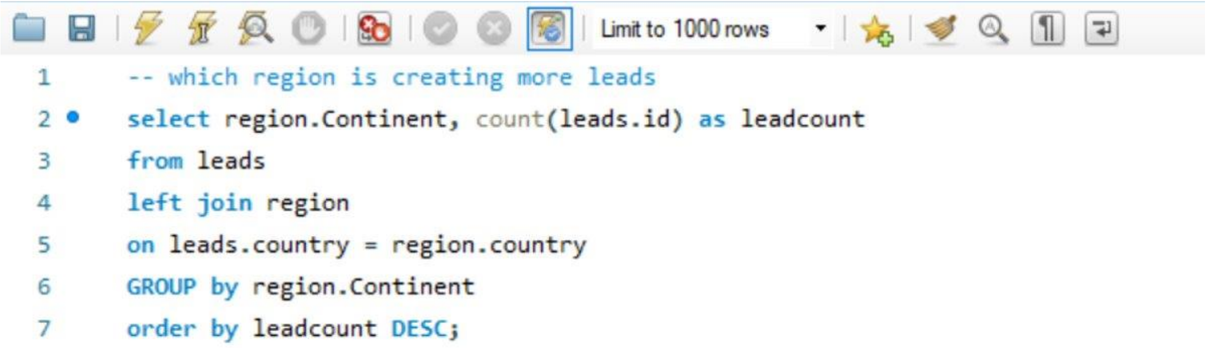
Output:



EmployeeID	leadmonth	leadcount
1	1	341
10	1	224
2	1	55
3	1	54
4	1	20
1	2	399
10	2	301
2	2	70
6	2	50
3	2	69
7	2	43

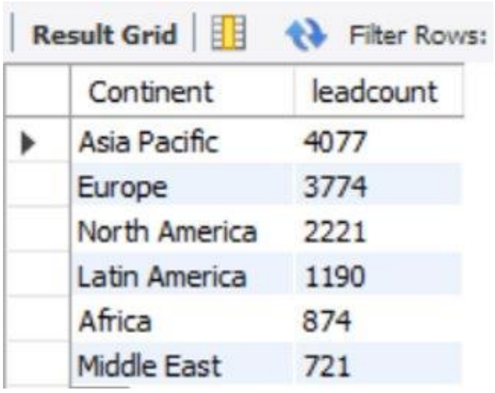
Business Application: The result of this query helps in understanding the month wise performance of each marketing member, which can help identify the best performers and also the consistency level in performances.

Objective 11 - To find the continent which generates most number of leads.



```sql
-- which region is creating more leads
select region.Continent, count(leads.id) as leadcount
from leads
left join region
on leads.country = region.country
GROUP by region.Continent
order by leadcount DESC;
```

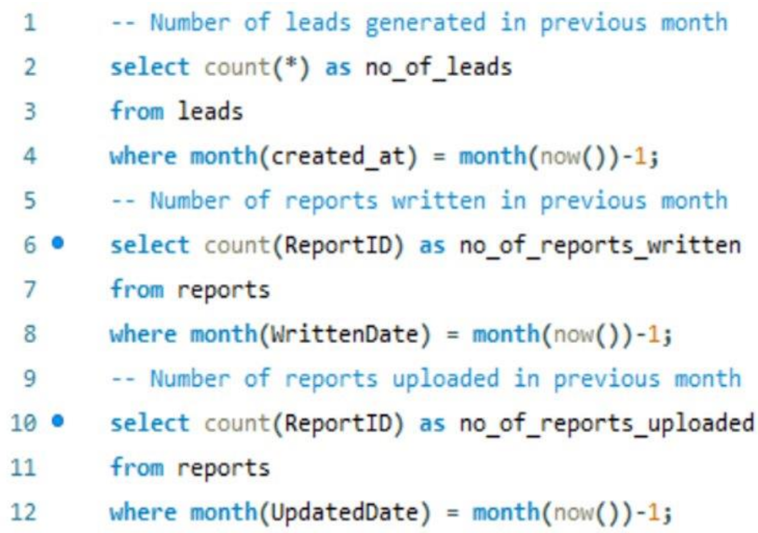
Output:



Continent	leadcount
Asia Pacific	4077
Europe	3774
North America	2221
Latin America	1190
Africa	874
Middle East	721

Business Application: The result of this query helps in identifying the region/continent with most number of leads and identify top attractive regions in terms of market potential and plan the marketing activities accordingly.

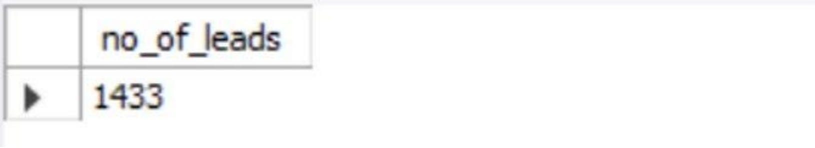
Objective 12 – Few summary KPI's that can help the organization.(group of queries)



```
-- Number of leads generated in previous month
select count(*) as no_of_leads
from leads
where month(created_at) = month(now())-1;
-- Number of reports written in previous month
select count(ReportID) as no_of_reports_written
from reports
where month(WrittenDate) = month(now())-1;
-- Number of reports uploaded in previous month
select count(ReportID) as no_of_reports_uploaded
from reports
where month(UpdatedDate) = month(now())-1;
```

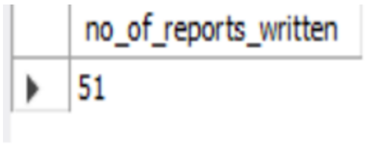
Output:

Number of leads generated in last month



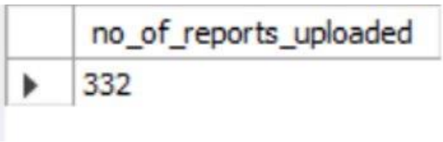
	no_of_leads
▶	1433

Number of reports written by content writers in last month



	no_of_reports_written
▶	51

Number of reports uploaded onto website in the past month



	no_of_reports_uploaded
▶	332

Recommendations:

Data Integrity and Validation:

- → Implement data validation checks to minimize manual data errors and ensure data accuracy.

Structured Storage:

- → Store all data in a structured format, adhering to the Third Normal Form (3NF), for easy reference in the future.

Scalability and Performance:

- → Design the database for computational efficiency, allowing it to handle large data volumes with minimal system lags.
- → Utilize appropriate indexing and partitioning strategies for performance optimization.

Data Volume:

- → Recognize that Excel is not suitable for storing extensive data sets due to limitations; an SQL database is better equipped for handling large amounts of data.
- → By following these recommendations, you can maintain data accuracy, structure, and performance, which Excel may struggle to handle for extensive data storage.