

SQL-Mongo Project – IBM HR Analytics Employee Attrition & Performance

BUAN 6320

Group Members

- 1) Sai Pranay Reddy Aare
- 2) Suman Akavaram
- 3) Keerti Namani
- 4) Indrasena Reddy Vukanti
- 5) Sreedhar Tatapudi

Group 20

Activity	Member 1	Member 2	Member 3	Member 4	Member 5
Prepared Data Model and Created Physical DB	Sreedhar				
Loaded Data into Database		Pranay			
Wrote SQL Queries		Pranay			Sourav
Prepared Mongo Database				Sumana	
Loaded data into Mongo DB			Indrasena Reddy		
Wrote Mongo Queries			Indrasena Reddy		
Prepared Report	Sreedhar			Sumana	
Reviewed Report					Sourav

Contents

Relational Data Model.....	5
Assumptions/Notes About Data Entities and Relationship	
Entity-Relationship Diagram	
Physical MySQL Database	
Assumptions/Notes About Data Set	
Screen shot of Physical Database objects	
Data in the Database	
SQL Queries.....	17
SQL Query 1	
Question	
Notes/Comments About SQL Query and Results (Include # of Rows in Result)	
Translation	
Screen Shot of SQL Query and Results	
SQL Query 2	
Question	
Notes/Comments About SQL Query and Results (Include # of Rows in Result)	
Translation	
Screen Shot of SQL Query and Results	
SQL Query 3	
Question	
Notes/Comments About SQL Query and Results (Include # of Rows in Result)	
Translation	
Screen Shot of SQL Query and Results	
SQL Query 4	
Question	
Notes/Comments About SQL Query and Results (Include # of Rows in Result)	
Translation	
Screen Shot of SQL Query and Results	
SQL Query 5	
Question	
Notes/Comments About SQL Query and Results (Include # of Rows in Result)	
Screen Shot of SQL Query and Results	

SQL Query 6	
Question	
Notes/Comments About SQL Query and Results (Include # of Rows in Result)	
Screen Shot of SQL Query and Results	
Data Review for MongoDB.....	25
Assumptions/Notes About Data Collections, Attributes and Relationships between Collections	
Physical Mongo Database	
Assumptions/Notes About Data Set	
Screen shot of Physical Database objects (Database, Collections and Attributes)	
Data in the Database	
MongoDB Queries/Code.....	29
Mongo Query 1	
Question	
Notes/Comments About MongoDB Query/Code and Results (Include # of Documents in Result)	
Translation	
Screen Shot of MongoDB Query/Code and Results	
Mongo Query 2	
Question	
Notes/Comments About MongoDB Query/Code and Results (Include # of Documents in Result)	
Translation	
Screen Shot of MongoDB Query/Code and Results	
Mongo Query 3	
Question	
Notes/Comments About MongoDB Query/Code and Results (Include # of Documents in Result)	
Translation	
Screen Shot of MongoDB Query/Code and Results	
Mongo Query 4	
Question	
Notes/Comments About MongoDB Query/Code and Results (Include # of Documents in Result)	
Translation	
Screen Shot of MongoDB Query/Code and Results	

Relational Data Model

Assumptions/Notes About Data Entities and Relationships

Entities:

- Salaries
- WorkExperience
- PerformanceandSalary
- DepartmentRoles
- PerformanceRating
- Employees
- Education
- RelationshipSatisfaction
- WorklifeBalance
- EmployeeReview
- JobInvolvement
- EnvironmentSatisfaction
- JobSatisfaction

Relationships:

- An employee can have only one salary (1:1) (Employees, Salaries)
- An employee can have only one Highest Education Level (1:1) (Employee, Education)
- An employee is associated with one work experience (1:1) (Employee, WorkExperience)
- Each employee has one review associated with him/her (1:1) (Employee, EmployeeReview)
- PerformanceandSalary has one PerformanceRating associated with it (1:1) (PerformanceandSalary, PerformanceRating)
- Each Employee review is associated with each individual work life balance of the employee (1:1)
- Each Employee review is associated with each individual relationship satisfaction of the employee(1:1) (Employee review, RelationshipSatisfaction)
- Each Employee review is associated with each individual environment satisfaction of the employee (1:1) (Employee review, EnvironmentSatisfaction)
- Each Employee review is associated with each individual job involvement of the employee (1:1)

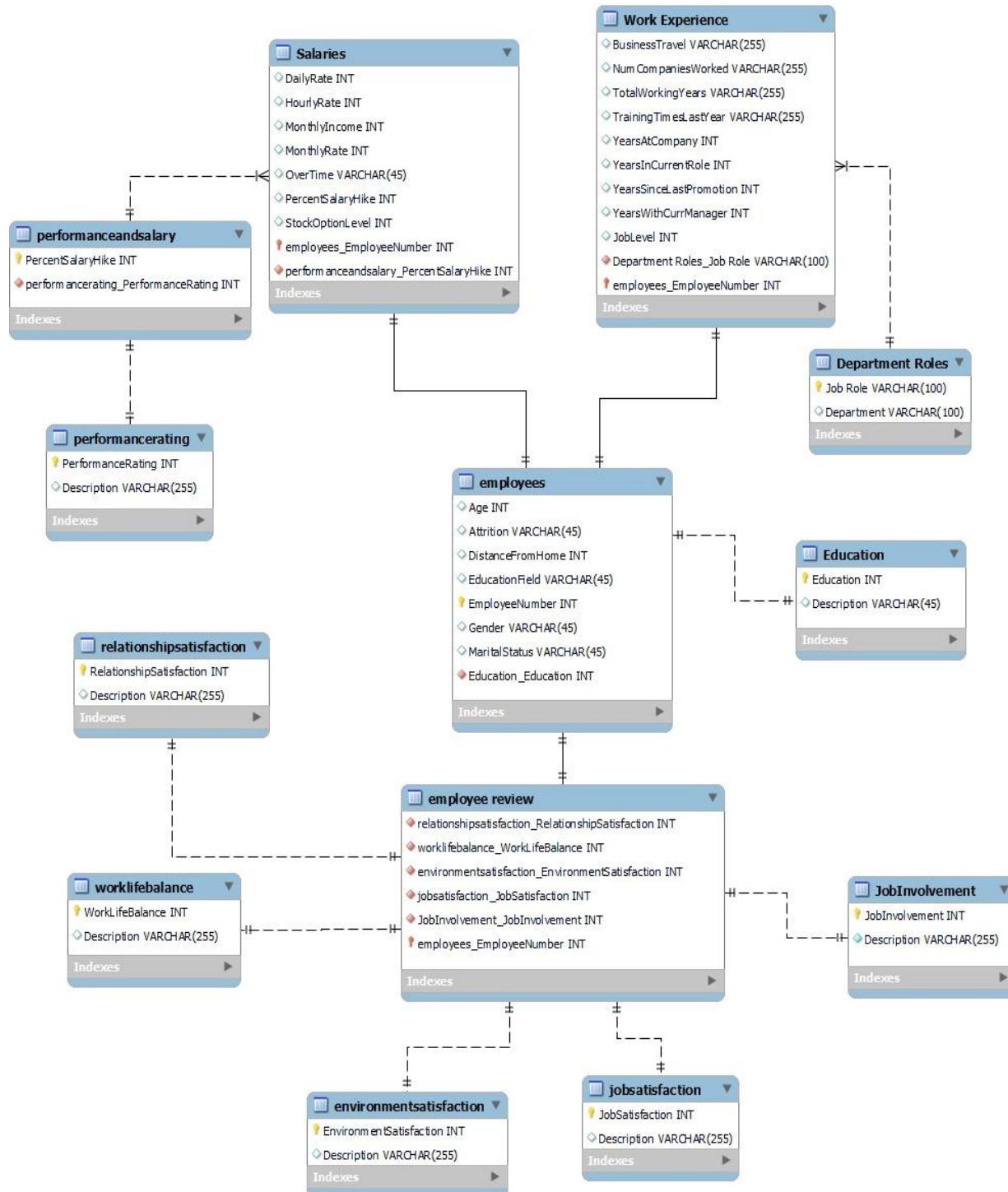
- Each Employee review is associated with each individual job satisfaction of the employee (1:1) (Employee review, JobSatisfaction)
- PerformanceandSalary and1 salaries have one to many relationship (1:n) (PerformanceandSalary, Salaries)
- DepartmentRoles and WorkExperience have one to many relationship(1:n) (Department Roles, WorkExperience)

Reasons why the data model is in 3NF:

- 1 NF – “EmployeeNumber” in the given data is unique, so 1 NF normality is achieved
- 2 NF - The data is divided into small entities in such a way that partial dependency is eliminated, i.e., all the fields in the entity is functionally dependent on the primary key
- 3 NF - The entities are created in such a way that the transitive dependency is eliminated i.e., every non key attribute is functionally dependent on the primary key and not on any other non key attribute

The “Department” depends on “Job Role” and “Job Role” depends on “EmployeeNumber”, we created the “DepartmentRoles” table such that “job role” depends only on “Department”. Transitive dependency is eliminated.

Entity-Relationship Diagram



Physical MySQL Database

Assumptions/Notes About Data Set

- We can decide if the age of the person is above or below 18 from "Age" column, so we removed "Over18" column
- "EmployeeCount" column is also removed because the row wise count of employee will be 1
- "StandardHours" column is also removed because it is same for all the employees
- "Manager" role is present across all the departments like, "Sales", "Human Resources", "Research and Development", In order to satisfy the primary key constraint, we replaced them with "Sales Manager", "Human Resources Manager", "Research and Development Manager", in the "Job Role" column while creating the "Department Role" table
- The table called "employees" contains all static columns related to an employee.
- Attrition indicates that the specific individual is no longer employed.
- Each department has unique roles within it that do not cross over with those of any other department.
- The number of trainings the employee completed with this firm in the previous year is shown by the field "TrainingTimesLastYear."
- In a "Employee Review," employees are asked to review their experiences at work and in their personal lives.
- The "Work Experience" Table includes all work-related experience, including employment level and role within the current organization, which can vary.
- All workers get monthly wages. The "hourly rate," "daily rate," and "monthly rate" refer to the compensation paid to contractors hired on the same day who are not part of the firm payroll. There is no listing of the contract workers here.

Screen shot of Physical Database objects

TABLES in “mydb” (“mydb” is the database created after we forward engineer the ER Diagram drawn for employee attrition dataset)

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

- SQL Editor:** Displays the following SQL code:

```
1 • USE mydb;
2 • SHOW TABLES;
```
- Result Grid:** Shows the list of tables found in the 'mydb' database:

Tables_in_mydb
department roles
education
employee review
employees
environmentsatisfaction
jobinvolvement
jobsatisfaction
performanceandsalary
performancerating
relationshipsatisfaction
salaries
work experience
worklifebalance
- Output Tab:** Shows the execution history of the queries:

#	Time	Action	Message	Duration / Fetch
186	20:55:35	SHOW TABLES	13 row(s) returned	0.015 sec / 0.000 sec
187	20:57:42	SELECT * FROM mydb.'department roles' LIMIT 0, 1000	11 row(s) returned	0.000 sec / 0.000 sec
188	21:06:53	EXPLAIN SELECT * FROM mydb.'department roles'	OK	0.000 sec
189	21:06:53	EXPLAIN FORMAT=JSON SELECT * FROM mydb.'department roles'	OK	0.000 sec
190	21:09:27	SELECT * FROM mydb.education LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
- Right Panel:** Displays a message about context help being disabled and instructions to use the toolbar for help.

“Department roles” table from “mydb”

The screenshot shows the MySQL Workbench interface with the 'department roles' tab selected. The results grid displays the following data:

Job Role	Department
Healthcare Representative	Research & Development
Human Resources	Human Resources
Human Resources Manager	Human Resources
Laboratory Technician	Research & Development
Manufacturing Director	Research & Development
Research & Development Manager	Research & Development
Research Director	Research & Development
Research Scientist	Research & Development
Sales Executive	Sales
Sales Manager	Sales

The SQL pane contains the query: `1 • SELECT * FROM mydb.department roles;`

The status bar at the bottom right indicates: "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help."

“Education” Table from mydb

The screenshot shows the MySQL Workbench interface with the 'education' tab selected. The results grid displays the following data:

Education	Description
1	Bellow College
2	College
3	Bachelor
4	Master
5	Doctor
HULL	HULL

The SQL pane contains the query: `1 • SELECT * FROM mydb.education;`

The status bar at the bottom right indicates: "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help."

"Review" Table from "mydb"

SQL Additions | SQL Additions

Limit to 1000 rows | Jump to

1 • SELECT * FROM mydb.`employee review`;

Automatic context help is disabled. Use the toolbar manually get help for the current caret position or toggle automatic help.

relationshipsatatisfaction_RelationshipSatisfaction	worklifebalance_WorkLifeBalance	environmentsatisfaction_EnvironmentSatisfaction	jobsatisfaction_JobSatisfaction	JobIr
1	1	2	4	3
4	3	3	2	2
2	3	4	3	2
3	3	4	3	3
4	3	1	2	3
3	2	4	4	3
1	2	3	1	4
2	3	4	3	3
2	3	4	3	2

loyee review 1 × Context Help Snippets

Action Output

#	Time	Action	Message	Duration / Fetch
190	21:09:27	SELECT * FROM mydb.education LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
191	21:27:31	SELECT * FROM mydb.education LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
192	21:36:58	SELECT * FROM mydb.`department roles` LIMIT 0, 1000	11 row(s) returned	0.000 sec / 0.000 sec
193	21:46:03	SELECT * FROM mydb.education LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
194	21:49:24	SELECT * FROM mydb.`employee review` LIMIT 0, 1000	1000 row(s) returned	0.015 sec / 0.000 sec

"Employees" table from "mydb"

SQL Additions | SQL Additions

Limit to 1000 rows | Jump to

1 • SELECT * FROM mydb.employees;

Automatic context help is disabled. Use the toolbar manually get help for the current caret position or toggle automatic help.

Age	Attrition	DistanceFromHome	EducationField	EmployeeNumber	Gender	MaritalStatus	Education_Education
41	Yes	1	Life Sciences	1	Female	Single	2
49	No	8	Life Sciences	2	Male	Married	1
37	Yes	2	Other	4	Male	Single	2
33	No	3	Life Sciences	5	Female	Married	4
27	No	2	Medical	7	Male	Married	1
32	No	2	Life Sciences	8	Male	Single	2
59	No	3	Medical	10	Female	Married	3
30	No	24	Life Sciences	11	Male	Divorced	1
38	No	23	Life Sciences	12	Male	Single	3
36	No	27	Medical	13	Male	Married	3

employees 1 × Context Help Snippets

Action Output

#	Time	Action	Message	Duration / Fetch
191	21:27:31	SELECT * FROM mydb.education LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
192	21:36:58	SELECT * FROM mydb.`department roles` LIMIT 0, 1000	11 row(s) returned	0.000 sec / 0.000 sec
193	21:46:03	SELECT * FROM mydb.education LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
194	21:49:24	SELECT * FROM mydb.`employee review` LIMIT 0, 1000	1000 row(s) returned	0.015 sec / 0.000 sec
195	21:52:29	SELECT * FROM mydb.employees LIMIT 0, 1000	1000 row(s) returned	0.016 sec / 0.000 sec

“Environment Satisfaction” table from “mydb”

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

- Navigator:** Shows the database schema. Under the 'mydb' schema, the 'environmentsatisfaction' table is selected.
- SQL Editor:** Displays the query: `1 • SELECT * FROM mydb.environmentsatisfaction;`. The results grid shows the following data:

EnvironmentSatisfaction	Description
1	Low
2	Medium
3	High
4	Very High
5	N/A

- Output Window:** Shows the history of actions taken on the table:

#	Time	Action	Message	Duration / Fetch
50	01:11:46	DROP TABLE mydb.environmentsatisfaction	Error Code: 3730. Cannot drop table 'environmentsatisfaction' referenced by a foreign key constraint f...	0.000 sec
51	01:17:07	SELECT * FROM mydb.environmentsatisfaction	4 row(s) returned	0.000 sec / 0.000 sec
52	01:18:18	UPDATE mydb.environmentsatisfaction SET Description='Medium' WHERE EnvironmentSatisfaction=...	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.031 sec
53	01:18:31	SELECT * FROM mydb.environmentsatisfaction	4 row(s) returned	0.016 sec / 0.000 sec
54	01:21:46	SELECT * FROM mydb.employee_review	1470 row(s) returned	0.000 sec / 0.000 sec
55	01:21:48	SELECT * FROM mydb.environmentsatisfaction	4 row(s) returned	0.000 sec / 0.000 sec
- System Bar:** Shows the system tray with icons for battery, signal, volume, and date/time (2:37 PM, 11/29/2022).

"JobInvolvement" table from "mydb"

SQL File 2 SQL File 3* education department roles education employee review employees environmentsatisfaction jobinvolvement jobinvolvement SQLAdditions

1 • **SELECT * FROM mydb.jobinvolvement;**

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Result Grid

JobInvolvement	Description
1	Low
2	Medium
3	High
4	Very High
*	NULL

involve 1 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
191	21:27:31	SELECT * FROM mydb.education LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
192	21:36:58	SELECT * FROM mydb.department roles' LIMIT 0, 1000	11 row(s) returned	0.000 sec / 0.000 sec
193	21:46:03	SELECT * FROM mydb.education LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
194	21:49:24	SELECT * FROM mydb.employee review' LIMIT 0, 1000	1000 row(s) returned	0.015 sec / 0.000 sec
195	21:52:29	SELECT * FROM mydb.employees LIMIT 0, 1000	1000 row(s) returned	0.016 sec / 0.000 sec
196	21:56:56	SELECT * FROM mydb.environmentsatisfaction LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
197	22:00:49	SELECT * FROM mydb.jobinvolvement LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec

"JobSatisfaction" table from "mydb"

SQL File 3* education department roles education employee review employees environmentsatisfaction jobinvolvement jobsatisfaction jobsatisfaction SQLAdditions

1 • **SELECT * FROM mydb.jobsatisfaction;**

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Result Grid

JobSatisfaction	Description
1	Low
2	Medium
3	High
4	Very High
*	NULL

bsatisfaction 1 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
192	21:36:58	SELECT * FROM mydb.department roles' LIMIT 0, 1000	11 row(s) returned	0.000 sec / 0.000 sec
193	21:46:03	SELECT * FROM mydb.education LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
194	21:49:24	SELECT * FROM mydb.employee review' LIMIT 0, 1000	1000 row(s) returned	0.015 sec / 0.000 sec
195	21:52:29	SELECT * FROM mydb.employees LIMIT 0, 1000	1000 row(s) returned	0.016 sec / 0.000 sec
196	21:56:56	SELECT * FROM mydb.environmentsatisfaction LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
197	22:00:49	SELECT * FROM mydb.jobinvolvement LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
198	22:03:14	SELECT * FROM mydb.jobsatisfaction LIMIT 0, 1000	4 row(s) returned	0.015 sec / 0.000 sec

"PerformanceandSalary" table from "mydb"

1 • `SELECT * FROM mydb.performanceandsalary;`

PercentSalaryHike	performanceRating_PerformanceRating
11	3
12	3
13	3
14	3
15	3
16	3
17	3
18	3
?	?

Output:

Action Output

#	Time	Action	Message	Duration / Fetch
193	21:46:03	SELECT * FROM mydb.education LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
194	21:49:24	SELECT * FROM mydb.employee_review LIMIT 0, 1000	1000 row(s) returned	0.015 sec / 0.000 sec
195	21:52:29	SELECT * FROM mydb.employees LIMIT 0, 1000	1000 row(s) returned	0.016 sec / 0.000 sec
196	21:56:56	SELECT * FROM mydb.environmentsatisfaction LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
197	22:00:49	SELECT * FROM mydb.jobinvolvement LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
198	22:03:14	SELECT * FROM mydb.jobsatisfaction LIMIT 0, 1000	4 row(s) returned	0.015 sec / 0.000 sec
199	22:06:27	SELECT * FROM mydb.performanceandsalary LIMIT 0, 1000	19 row(s) returned	0.000 sec / 0.000 sec

"Performancerating" table in "mydb"

1 • `SELECT * FROM mydb.performancerating;`

PerformanceRating	Description
1	Low
2	Good
3	Excellent
4	Outstanding
*	HULL

Output:

Action Output

#	Time	Action	Message	Duration / Fetch
194	21:49:24	SELECT * FROM mydb.employee_review LIMIT 0, 1000	1000 row(s) returned	0.015 sec / 0.000 sec
195	21:52:29	SELECT * FROM mydb.employees LIMIT 0, 1000	1000 row(s) returned	0.016 sec / 0.000 sec
196	21:56:56	SELECT * FROM mydb.environmentsatisfaction LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
197	22:00:49	SELECT * FROM mydb.jobinvolvement LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
198	22:03:14	SELECT * FROM mydb.jobsatisfaction LIMIT 0, 1000	4 row(s) returned	0.015 sec / 0.000 sec
199	22:06:27	SELECT * FROM mydb.performanceandsalary LIMIT 0, 1000	19 row(s) returned	0.000 sec / 0.000 sec
200	22:11:13	SELECT * FROM mydb.performancerating LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec

“relationshipsatisfaction” from “mydb”

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

- Toolbar:** Employee review, employees, environmentsatisfaction, jobinvolvement, jobsatisfaction, performanceandsalary, performancerating, relationships.
- SQL Editor:** The query `1• SELECT * FROM mydb.relationshipsatisfaction;` is entered.
- Result Grid:** Displays the data from the relationshipsatisfaction table:

RelationshipSatisfaction	Description
1	Low
2	Medium
3	High
4	Very High
*	NULL
- Output Panel:** Shows the execution history of the session, including the current query and previous SELECT statements on other tables.
- Help:** A note on the right says: "Automatic context help is disabled. Use the toolbar or manually get help for the current caret position or toggle automatic help."

“Salaries” table from “mydb”

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

- Toolbar:** environmentsatisfaction, jobinvolvement, jobsatisfaction, performanceandsalary, performancerating, relationshipsatisfaction, salaries.
- SQL Editor:** The query `1• SELECT * FROM mydb.salaries;` is entered.
- Result Grid:** Displays the data from the salaries table:

DailyRate	HourlyRate	MonthlyIncome	MonthlyRate	Overtime	PercentSalaryHike	StockOptionLevel	employees_EmployeeNumber	performanceandsalary_Percent
1102	94	5993	19479	Yes	11	0	1	11
279	61	5130	24907	No	23	1	2	23
1373	92	2090	2396	Yes	15	0	4	15
1392	56	2909	23159	Yes	11	0	5	11
591	40	3468	16632	No	12	1	7	12
1005	79	3068	11864	No	13	0	8	13
1324	81	2670	9964	Yes	20	3	10	20
1280	47	2602	12225	No	22	1	11	22
- Output Panel:** Shows the execution history of the session, including the current query and previous SELECT statements on other tables.
- Help:** A note on the right says: "Automatic context help is disabled. Use the toolbar or manually get help for the current caret position or toggle automatic help."

"Work Experience" table from "mydb"

The screenshot shows the MySQL Workbench interface with the 'work experience' table selected. The table has columns: BusinessTravel, NumCompaniesWorked, TotalWorkingYears, TrainingTimesLastYear, YearsAtCompany, YearsInCurrentRole, YearsSinceLastPromotion, and YearsWithIT. The data shows various travel frequencies and years of experience.

BusinessTravel	NumCompaniesWorked	TotalWorkingYears	TrainingTimesLastYear	YearsAtCompany	YearsInCurrentRole	YearsSinceLastPromotion	YearsWithIT
Travel_Rarely	8	8	0	6	4	0	5
Travel_Frequently	1	10	3	10	7	1	7
Travel_Rarely	6	7	3	0	0	0	0
Travel_Frequently	1	8	3	8	7	3	0
Travel_Rarely	9	6	3	2	2	2	2
Travel_Frequently	0	8	2	7	7	3	6
Travel_Rarely	4	12	3	1	0	0	0

The 'Output' pane shows the execution history:

```

197 22:00:49 SELECT * FROM mydb.jobinvolvement LIMIT 0, 1000
198 22:03:14 SELECT * FROM mydb.jobsatisfaction LIMIT 0, 1000
199 22:06:27 SELECT * FROM mydb.performanceandsalary LIMIT 0, 1000
200 22:11:13 SELECT * FROM mydb.performancerating LIMIT 0, 1000
201 22:13:39 SELECT * FROM mydb.relationshipsatisfaction LIMIT 0, 1000
202 22:16:28 SELECT * FROM mydb.salaries LIMIT 0, 1000
203 22:20:28 SELECT * FROM mydb.'work experience' LIMIT 0, 1000
  
```

"Work life balance" table from "mydb"

The screenshot shows the MySQL Workbench interface with the 'worklifebalance' table selected. The table has columns: WorkLifeBalance and Description. The data shows four levels of work-life balance: Bad, Good, Better, and Best.

WorkLifeBalance	Description
1	Bad
2	Good
3	Better
4	Best
NULL	NULL

The 'Output' pane shows the execution history:

```

198 22:03:14 SELECT * FROM mydb.jobsatisfaction LIMIT 0, 1000
199 22:06:27 SELECT * FROM mydb.performanceandsalary LIMIT 0, 1000
200 22:11:13 SELECT * FROM mydb.performancerating LIMIT 0, 1000
201 22:13:39 SELECT * FROM mydb.relationshipsatisfaction LIMIT 0, 1000
202 22:16:28 SELECT * FROM mydb.salaries LIMIT 0, 1000
203 22:20:28 SELECT * FROM mydb.'work experience' LIMIT 0, 1000
204 22:23:41 SELECT * FROM mydb.worklifebalance LIMIT 0, 1000
  
```

Data in the Database

Table Name	Primary Key	Foreign Key	# of Rows in Table
Department roles	Job role		11
Education	Education		5
Employee Review	EmployeeNumber	EmployeeNumber EmployeeSatisfaction JobInvolvement JobSatisfaction RelationshipSatisfaction WorkLifeBalance	1470
Employees	EmployeeNumber	Education	1470
EnvironmentSatisfaction	EnvironmentSatisfaction		4
JobInvolvement	JobInvolvement		4
JobSatisfaction	JobSatisfaction		4
Performanceandsalary	PercentSalaryHike	PerformanceRating	19
Performance Rating	PerformanceRating		4
Salaries	EmployeeNumber	EmployeeNumber PercentSalaryHike	1470
WorkExperience	EmployeeNumber	EmployeeNumber Job Role	1470
Worklifebalance	WorkLifeBalance		4

Note: if we see the tables “Employee Review”, “Employees”, “Salaries”, “WorkExperience”, the number of rows returned are 1470 according to the above table, but from the screenshots we provided, the number of rows returned are 1000, this is due to LIMIT constraint set default to 1000 by MySQL. we can uncheck this constraint and make mysql to return 1470 rows of that particular table.

This can be done by

Edit → Preferences → SQL Queries → Limit Rows → Set COUNT to 10000 or Uncheck “Limit Rows” → click “OK”

SQL Queries

Pick 5 out of the 12 statements to write your queries in MySQL.

SQL Query 1

3. A new employee from a Medical-related education field wants to work in Sales. Do you believe the company might be able to give her a chance to work in Sales? Why or Why not?

Assumptions:

1. Assume that new employee is able to work in sales team. We need to check if there are any female employees who are working in sales department with Medical related education field. If there are employees related to that field then she can work in that department.
2. The query written below has the output which has employees with medical related education field in all departments

Code:

```
select distinct `department` from mydb.`department roles`  
where `Job Role` in (select `Department Roles_Job Role` From mydb.`work experience` where  
'employees_EmployeeNumber' in (select `EmployeeNumber` from mydb.`employees`  
where `EducationField`="Medical" and `Gender`="Female"));
```

Output:

The screenshot shows the SSMS interface with the following details:

- Query Editor:** Displays the T-SQL query:

```
1 use mydb;  
2  
3 • select distinct `department` from mydb.`department roles`  
4 where `Job Role` in (select `Department Roles_Job Role` From mydb.`work experience` where  
5 `employees_EmployeeNumber` in (select `EmployeeNumber` from mydb.`employees`  
6 where `EducationField`="Medical" and `Gender`="Female"));  
7
```

- Result Grid:** Shows the output of the query:

department
Research & Development
Human Resources
Sales
- SQL Additions:** A sidebar with the message: "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help."
- Action Output:** A table showing the execution log with the following data:

#	Time	Action	Message	Duration / Fetch
140	19:59:29	Select count(*) as Total_BusinessTravel,count(BusinessTravel) From mydb.`work experience` Group by ...	3 row(s) returned	0.000 sec / 0.000 sec
141	20:00:10	Select BusinessTravel,count(BusinessTravel)/1470 From mydb.`work experience` Group by BusinessTra...	3 row(s) returned	0.000 sec / 0.000 sec
142	20:17:07	select distinct `department` from mydb.`department roles` where `Job Role` in (select `Department Roles_...	3 row(s) returned	0.016 sec / 0.000 sec
143	20:17:15	SELECT * FROM mydb.`department roles`	11 row(s) returned	0.016 sec / 0.000 sec
144	20:25:58	select distinct `department` from mydb.`department roles` where `Job Role` in (select `Department Roles_...	3 row(s) returned	0.016 sec / 0.000 sec

The above output shows the female employees in all departments who has medical related education field. The sales department has employees who has medical related education background. If employees are working, then this new female employee will also be able to work in Sales department. Yes, I believe that company will give a chance for the female employee who has medical related education background to work.

SQL Query 2

9. A new employee from a Life Sciences education field wants to work in Sales. Do you believe the company might be able to give him a chance to work in Sales? Why or Why not?

1. Assume that new employee is able to work in sales team. We need to check if there are any male employees who are working in sales department with Life Sciences education field.
If there are employees related to that field then she can work in that department.
2. The query written below has the output which has employees with Life sciences education field in all departments

Code:

```
select distinct `department` from mydb.`department roles`  
where `Job Role` in (select `Department Roles_Job Role` From mydb.`work experience` where  
'employees_EmployeeNumber' in (select `EmployeeNumber` from mydb.`employees`  
where `EducationField`="Life Sciences" and `Gender`="Male"));
```

Output:

The screenshot shows the SSMS interface with the following details:

- Query Editor:** Displays the T-SQL query:

```
7  
8 • select distinct `department` from mydb.`department roles`  
9   where `Job Role` in (select `Department Roles_Job Role` From mydb.`work experience` where  
10  `employees_EmployeeNumber` in (select `EmployeeNumber` from mydb.`employees`  
11    where `EducationField`="Life Sciences" and `Gender`="Male"));  
12  
13 • select distinct `department` from mydb.`department roles`
```

- Result Grid:** Shows the output of the query:

department
Research & Development
Human Resources
Sales

- Message List:** Shows the execution log:

#	Time	Action	Message	Duration / Fetch
141	20:00:10	Select BusinessTravel.count(BusinessTravel)/1470 From mydb.`work experience` Group by BusinessTr...	3 row(s) returned	0.000 sec / 0.000 sec
142	20:17:07	select distinct `department` from mydb.`department roles` where `Job Role` in (select `Department Roles_...	3 row(s) returned	0.016 sec / 0.000 sec
143	20:17:15	SELECT * FROM mydb.`department roles`	11 row(s) returned	0.016 sec / 0.000 sec
144	20:25:58	select distinct `department` from mydb.`department roles` where `Job Role` in (select `Department Roles_...	3 row(s) returned	0.016 sec / 0.000 sec
145	20:38:13	select distinct `department` from mydb.`department roles` where `Job Role` in (select `Department Roles_...	3 row(s) returned	0.000 sec / 0.000 sec

The above output shows the male employees in all departments who has Life Sciences education field. The sales department has employees who has Life Sciences education background. If employees are working, then this new male employee will also be able to work in Sales department.
Yes, I believe that company will give a chance for the male employee who has Life Sciences education background to work.

SQL Query 3

8.The company has been paying gas expenses for miles traveled by employees between their home and work. If they want to increase the per mile compensation, which department's employees will gain the most?

Approach:

For this question, we have calculated the total distance from work for all employees by department. We calculated sum of all distances of employees by department. So, we will know how many miles travelled by employees of each department. The department which has highest total distance travelled will have more expenses as miles travelled increases.

Query:

```
select department, sum(`distancefromhome`) as `Distance` from mydb.`employees` as e
Join mydb.`work experience` as w
on e.EmployeeNumber=w.employees_EmployeeNumber
Join mydb.`department roles` as d
on w.`Department Roles_Job Role`= d.`Job Role`
group by Department;
```

Output:

The screenshot shows the SQL Server Management Studio interface. The top menu bar includes 'Query 1', 'SQL File 6*', 'education', 'employee review', 'SQL File 7*', 'work experience', and 'department roles'. The toolbar contains various icons for file operations, search, and help. A status bar at the bottom right shows the date and time: '8:49 PM 12/10/2022'.

The main area displays the following SQL code:

```
19
20 • select department, sum(`distancefromhome`) as `Distance` from mydb.`employees` as e
21   Join mydb.`work experience` as w
22   on e.EmployeeNumber=w.employees_EmployeeNumber
23   Join mydb.`department roles` as d
24   on w.`Department Roles_Job Role`= d.`Job Role`
25   group by Department;
26
27 • select * From mydb.`work experience` as w
28   Join mydb.`department roles` as d
29   on w.`Department Roles Job Role` = d.`Job Role`
```

The 'Result Grid' tab is selected, showing the results of the query:

department	Distance
Research & Development	8788
Human Resources	548
Sales	4177

The 'Action Output' tab shows the execution log:

#	Time	Action	Message	Duration / Fetch
142	20:17:07	select distinct 'department' from mydb.`department roles` where 'Job Role' in (select 'Department Roles....'	3 row(s) returned	0.016 sec / 0.000 sec
143	20:17:15	SELECT 'FROM mydb.`department roles`'	11 row(s) returned	0.016 sec / 0.000 sec
144	20:25:58	select distinct 'department' from mydb.`department roles` where 'Job Role' in (select 'Department Roles....'	3 row(s) returned	0.016 sec / 0.000 sec
145	20:38:13	select distinct 'department' from mydb.`department roles` where 'Job Role' in (select 'Department Roles....'	3 row(s) returned	0.000 sec / 0.000 sec
146	20:48:23	select department, sum(`distancefromhome`) as `Distance` from mydb.`employees` as e Join mydb.`work ...	3 row(s) returned	0.015 sec / 0.000 sec

From the above output, the Research and Development department employees has travelled more distance than Human resources and sales. So, the expenses for gas will be more for Research and development department.

The Research &Development department employees has 8788 miles distance.

SQL Query 4

4. The HR department feels they have the highest job satisfaction while Research & Development department feels their department has the highest environment satisfaction. Who is right?

Approach:

We had calculated the proportion of employees who have very high job satisfaction and very high environment satisfaction in each department.

We calculated this job satisfaction and environment satisfaction rating proportions for all departments by grouping them.

Query:

```
select x.Department, x.`Job satisfaction`/y.`Total Job satisfaction` as `Job satisfaction`,  
x.`Environment satisfaction`/y.`Total Environment satisfaction` as `Environment satisfaction`  
from  
(select a.Department, `Job satisfaction`, `Environment satisfaction`  
from  
(select Department, count(jobsatisfaction.Description) as `Job satisfaction`  
from `department roles`  
join `work experience`  
on `department roles`.`Job Role` = `work experience`.`Department Roles_Job Role`  
join `employee review`  
on `work experience`.`employees_EmployeeNumber` = `employee  
review`.`employees_EmployeeNumber`  
join Jobsatisfaction  
on `employee review`.`jobsatisfaction_JobSatisfaction` = Jobsatisfaction.JobSatisfaction  
where jobsatisfaction.Description = 'Very High'  
group by Department)a  
join  
(select Department, count(environmentsatisfaction.Description) as `Environment satisfaction`  
from `department roles`  
join `work experience`  
on `department roles`.`Job Role` = `work experience`.`Department Roles_Job Role`  
join `employee review`  
on `work experience`.`employees_EmployeeNumber` = `employee  
review`.`employees_EmployeeNumber`  
join environmentsatisfaction  
on `employee review`.`environmentsatisfaction_EnvironmentSatisfaction` =  
environmentsatisfaction.EnvironmentSatisfaction  
where environmentsatisfaction.Description = 'Very High'  
group by Department)b  
on a.Department = b.Department) as x  
join  
(select a.Department, `Total Job satisfaction`, `Total Environment satisfaction`  
from  
(select Department, count(jobsatisfaction.Description) as `Total Job satisfaction`  
from `department roles`  
join `work experience`  
on `department roles`.`Job Role` = `work experience`.`Department Roles_Job Role`
```

```

join `employee review`
on `work experience`.`employees_EmployeeNumber` = `employee
review`.`employees_EmployeeNumber`
join Jobsatisfaction
on `employee review`.`jobsatisfaction_JobSatisfaction` = Jobsatisfaction.JobSatisfaction
group by Department) as a
join
(select Department, count(environmentsatisfaction.Description) as `Total Environment satisfaction`
from `department roles`
join `work experience`
on `department roles`.`Job Role` = `work experience`.`Department Roles_Job Role`
join `employee review`
on `work experience`.`employees_EmployeeNumber` = `employee
review`.`employees_EmployeeNumber`
join environmentsatisfaction
on `employee review`.`environmentsatisfaction_EnvironmentSatisfaction`=
environmentsatisfaction.EnvironmentSatisfaction
group by Department) as b
on a.Department = b.Department) as y
on x.Department = y.Department;

```

Output:

The screenshot shows the SQL Server Management Studio interface with the following details:

- Query Editor:** Displays the T-SQL query used to calculate job satisfaction.
- Result Grid:** Shows a table with three columns: Department, Job satisfaction, and Environment satisfaction. The data is as follows:

Department	Job satisfaction	Environment satisfaction
Sales	0.3296	0.2848
Research & Development	0.3070	0.3174
Human Resources	0.2698	0.2222

- Output Window:** Displays the execution log with the following entries:

#	Action	Time	Message	Duration / Fetch
142	select distinct 'department' from mydb.'department roles' where 'Job Role' in (select 'Department Roles.....'	20:17:07	3 row(s) returned	0.016 sec / 0.000 sec
143	SELECT * FROM mydb.'department roles'	20:17:15	11 row(s) returned	0.016 sec / 0.000 sec
144	select distinct 'department' from mydb.'department roles' where 'Job Role' in (select 'Department Roles.....'	20:25:58	3 row(s) returned	0.016 sec / 0.000 sec
145	select distinct 'department' from mydb.'department roles' where 'Job Role' in (select 'Department Roles.....'	20:38:13	3 row(s) returned	0.000 sec / 0.000 sec
146	select department, sum('distancefromhome') as 'Distance' from mydb.'employees' as e Join mydb.'work	20:48:23	3 row(s) returned	0.015 sec / 0.000 sec

From the above output, we observe that Environment satisfaction rating is high for Research & Development than that of Human Resources and Sales.

We also observed that Job satisfaction rating is high for Sales than that of Research & Development and Human Resources. The question states that HR department feels they have the highest job satisfaction

while Research & Development department feels their department has the highest environment satisfaction but we got highest job satisfaction rating for Sales.

So only one statement by R&D is correct regarding environment satisfaction which is high for Research & development department.

The statement by HR department is incorrect because sales have highest job satisfaction.

SQL Query 5

10. HR feels that their environment satisfaction score is higher than Sales but HR job satisfaction score is lower than Research & Development. Are they right?

Approach:

We had calculated the proportion of employees who have very high job satisfaction and very high environment satisfaction in each department.

We calculated this job satisfaction and environment satisfaction rating proportions for all departments by grouping them.

Query:

```
select x.Department, x.`Job satisfaction`/y.`Total Job satisfaction` as `Job satisfaction`,  
x.`Environment satisfaction`/y.`Total Environment satisfaction` as `Environment satisfaction`  
from  
(select a.Department, `Job satisfaction`, `Environment satisfaction`  
from  
(select Department, count(jobsatisfaction.Description) as `Job satisfaction`  
from `department roles`  
join `work experience`  
on `department roles`.`Job Role` = `work experience`.`Department Roles_Job Role`  
join `employee review`  
on `work experience`.`employees_EmployeeNumber` = `employee  
review`.`employees_EmployeeNumber`  
join Jobsatisfaction  
on `employee review`.`jobsatisfaction_JobSatisfaction` = Jobsatisfaction.JobSatisfaction  
where jobsatisfaction.Description = 'Very High'  
group by Department)a  
join  
(select Department, count(environmentsatisfaction.Description) as `Environment satisfaction`  
from `department roles`  
join `work experience`  
on `department roles`.`Job Role` = `work experience`.`Department Roles_Job Role`  
join `employee review`  
on `work experience`.`employees_EmployeeNumber` = `employee  
review`.`employees_EmployeeNumber`  
join environmentsatisfaction  
on `employee review`.`environmentsatisfaction_EnvironmentSatisfaction` =  
environmentsatisfaction.EnvironmentSatisfaction  
where environmentsatisfaction.Description = 'Very High'  
group by Department)b  
on a.Department = b.Department) as x  
join
```

```

(select a.Department, `Total Job satisfaction`, `Total Environment satisfaction`
from
(select Department, count(jobsatisfaction.Description) as `Total Job satisfaction`
from `department roles`
join `work experience`
on `department roles`.`Job Role` = `work experience`.`Department Roles_Job Role`
join `employee review`
on `work experience`.`employees_EmployeeNumber` = `employee
review`.`employees_EmployeeNumber`
join Jobsatisfaction
on `employee review`.jobsatisfaction_JobSatisfaction` = Jobsatisfaction.JobSatisfaction
group by Department) as a
join
(select Department, count(environmentsatisfaction.Description) as `Total Environment satisfaction`
from `department roles`
join `work experience`
on `department roles`.`Job Role` = `work experience`.`Department Roles_Job Role`
join `employee review`
on `work experience`.`employees_EmployeeNumber` = `employee
review`.`employees_EmployeeNumber`
join environmentsatisfaction
on `employee review`.environmentsatisfaction_EnvironmentSatisfaction`=
environmentsatisfaction.EnvironmentSatisfaction
group by Department) as b
on a.Department = b.Department) as y
on x.Department = y.Department;

```

Output:

The screenshot shows a SQL query editor interface with the following details:

- Query Editor:** Displays the executed SQL code. The code is a multi-step query involving joins between 'department roles', 'work experience', 'employee review', and 'Jobsatisfaction' tables to calculate total job and environment satisfaction per department, then joins these results to find departments with similar satisfaction levels.
- Result Grid:** Shows the output of the query as a table. The columns are 'Department', 'Job satisfaction', and 'Environment satisfaction'. The data rows are:

Department	Job satisfaction	Environment satisfaction
Sales	0.3296	0.2848
Research & Development	0.3070	0.3174
Human Resources	0.2698	0.2222
- SQL Additions:** A panel on the right provides context help, stating: "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help."
- Action Output:** A log at the bottom shows the execution history with details like time, action, message, and duration.

```

1 • select x.Department, x.`Job satisfaction`/y.`Total Job satisfaction` as `Job satisfaction`,
2   x.`Environment satisfaction`/y.`Total Environment satisfaction` as `Environment satisfaction`
3   from
4   (select a.Department, `Job satisfaction`, `Environment satisfaction`
5    from
6    (select Department, count(jobsatisfaction.Description) as `Total Job satisfaction`
7     from `department roles`
8     join `work experience`
9     on `department roles`.`Job Role` = `work experience`.`Department Roles_Job Role`
10    join `employee review`
11    on `work experience`.`employees_EmployeeNumber` = `employee
12    review`.`employees_EmployeeNumber`
13    join Jobsatisfaction
14
15 Result 3 x
16 Output
17 Action Output
18 # Time Action Message Duration / Fetch
19 142 20:17:07 select distinct 'department' from mydb.'department roles' where 'Job Role' in (select 'Department Roles_...
20 143 20:17:15 SELECT * FROM mydb.'department roles'
21 144 20:25:59 select distinct 'department' from mydb.'department roles' where 'Job Role' in (select 'Department Roles_...
22 145 20:38:13 select distinct 'department' from mydb.'department roles' where 'Job Role' in (select 'Department Roles_...
23 146 20:48:23 select department, sum('distancefromhome') as 'Distance' from mydb.'employees' as e Join mydb.'work ...

```

From the above output, we observe that Environment satisfaction rating is high for Research & Development than that of Human Resources and Sales.

We also observed that Job satisfaction rating is high for Sales than that of Research & Development and Human Resources.

The question states that HR feels that their environment satisfaction score is higher than Sales but HR job satisfaction score is lower than Research & Development

It is incorrect because sales have high environment satisfaction than HR

It is correct about job satisfaction because job satisfaction is higher for R&D than HR

SQL Query 6

Q1) Assume that the company has several branch offices around the country and employee morale is down. The company feels that not enough employees are traveling frequently between offices and that more employees should travel frequently to the branch offices to improve morale amongst each other. Is the company correct in feeling this way?

For the above question we calculated the count of employees by grouping BusinessTravel. The Business travel has 3 types: Travel_Rarely, Travel_Frequently and Non-Travel. We have written a query for count by grouping Businesstravel.

Query:

```
Select BusinessTravel,count(BusinessTravel)/1470 From mydb.'work experience'  
Group by BusinessTravel;
```

Output:

The screenshot shows the SQL Server Management Studio interface. The top pane displays the T-SQL query:

```
3   Inner Join mydb.'salaries'  
4   on mydb.'work experience'.'employees_EmployeeNumber'=mydb.'salaries'.'employees_EmployeeNumber'  
5   Group by 'BusinessTravel';  
6  
7 •  Select BusinessTravel,count(BusinessTravel)/1470 From mydb.'work experience'  
8   Group by BusinessTravel;  
9  
10  
11 •  Select 'BusinessTravel',Avg('performancerating_performancerating') from mydb.'work experience'
```

The bottom pane shows the result grid with the following data:

BusinessTravel	count(BusinessTravel)/1470
Travel_Rarely	0.7095
Travel_Frequently	0.1884
Non-Travel	0.1020

The status bar at the bottom right shows the time as 11:04 PM and the date as 12/10/2022.

From the output, we observed that 71% of employees are travelling rarely and 18.84% of employees travel frequently and 10.20% of employees do not travel at all.

From the above we can say that most of the employees are not travelling frequently between offices.

It is true the company feels that not enough employees are traveling frequently between offices and that more employees should travel frequently to the branch offices to improve morale amongst each other.

Data Review for MongoDB

Assumptions/Notes About Data Collections, Attributes and Relationships between Collections

- As MongoDB is Non-relational database, we can load the data directly in CSV format as it can handle the unstructured format
- Created the database “attritiondatabase”, with “employee_attrition” as collection in it. This is done using compass.
- Executed the queries in mongo shell.
- Removed the fields, “Over18”, “standard hours”, “EmployeeCount” from the original dataset
- The data is present in only single collection, so the relationships doesn’t exist
- Using the Mongodb tools in Atlas, we created the database and loaded the data using compass.
- Following are the details of cluster in Atlas

Username :ixv22001

Password: ABCabc12345

```
C:\Users\indra>cd C:\Users\indra\Downloads\mongosh-1.6.1-win32-x64\mongosh-1.6.1-win32-x64\bin
C:\Users\indra\Downloads\mongosh-1.6.1-win32-x64\mongosh-1.6.1-win32-x64\bin>mongosh "mongodb+srv://cluster20.oovrlm7.mongodb.net/myFirstDatabase" --apiVersion 1 --username ixv22001
Enter password: *****
Current Mongosh Log ID: 6394e5f9025ad3b477932173
Connecting to:      mongodb+srv://<credentials>@cluster20.oovrlm7.mongodb.net/myFirstDatabase?appName=mongosh+1.6.1
Using MongoDB:      5.0.14 (API Version 1)
Using Mongosh:      1.6.1
```

Physical Mongo Database

Assumptions/Notes About Data Set

- After importing the dataset, MongoDB implicitly identifies the unique field with “_id” column
- Attrition means that the person is no longer employed
- TrainingTimesLastYear are the number of trainings the employee has been assigned with the company in last year

Screen shot of Physical Database objects (Database, Collections and Attributes)

Screenshot of “attritiondatabase”

The screenshot shows the MongoDB Compass interface with the 'attritiondatabase' database selected. The 'Databases' tab is active, displaying four databases: admin, attritiondatabase, local, and employee_attrition. The 'attritiondatabase' database is highlighted. Each database card shows its storage size, collection count, and index count.

Database	Storage size	Collections	Indexes
admin	0 B	0	0
attritiondatabase	483.33 kB	1	1
local	-	-	-
employee_attrition	-	-	-

Screenshot of “employee_attrition” collection

The screenshot shows the MongoDB Compass interface with the 'attritiondatabase' database selected. The 'Collections' tab is active, displaying the 'employee_attrition' collection. The collection card provides detailed statistics: storage size, document count, average document size, index count, and total index size.

Collection	Storage size	Documents	Avg. document size	Indexes	Total index size
employee_attrition	286.72 kB	15 K	806.00 B	1	102.40 kB

Screenshot of “employee_attrition” collection with attributes

The screenshot shows the MongoDB Compass interface. The left sidebar lists databases: cluster20.0ovrlm7, admin, attritiondatabase, employee_attrition (selected), local, oplog.rs, and sample_training. The main area displays the 'attritiondatabase.employee_attrition' collection. It shows 1.5k DOCUMENTS and 1 INDEXES. A sample document is expanded, showing fields such as Age, BusinessTravel, Department, Education, Gender, HourlyRate, JobInvolvement, JobLevel, JobTitle, MaritalStatus, MonthlyIncome, MonthlyRate, NumCompaniesWorked, Overtime, PercentSalaryHike, PerformanceRating, and RelationshipSatisfaction. A button at the bottom says 'SHOW 8 MORE FIELDS'. The bottom status bar shows the system is running on MONGOSH, the temperature is 73°F, it's heavy rain, and the date and time are 12/10/2022 12:49 PM.

Data in the Database

Collection Name	Relationships With Other Collections (if any)	# of Documents in Collection
employee_attrition	NA	1470

MongoDB Queries/Code

Mongo Query 1

Question 2 : Which department's employee is most likely to have the shortest commute between Home and Work?

Approach:

Calculated the average distance travelled by an employee from each department. We found out the following details:

Average distance travelled by an employee from "Human Resources" department = 8.698

Average distance travelled by an employee from "Research and Development" department= 9.144

Average distance travelled by an employee from "Sales" department= 9.3654

So, "Human Resources" department's employee can have the shortest commute between home and work

In Addition to this, we have also calculated the count of employees from each department

Notes/Comments:

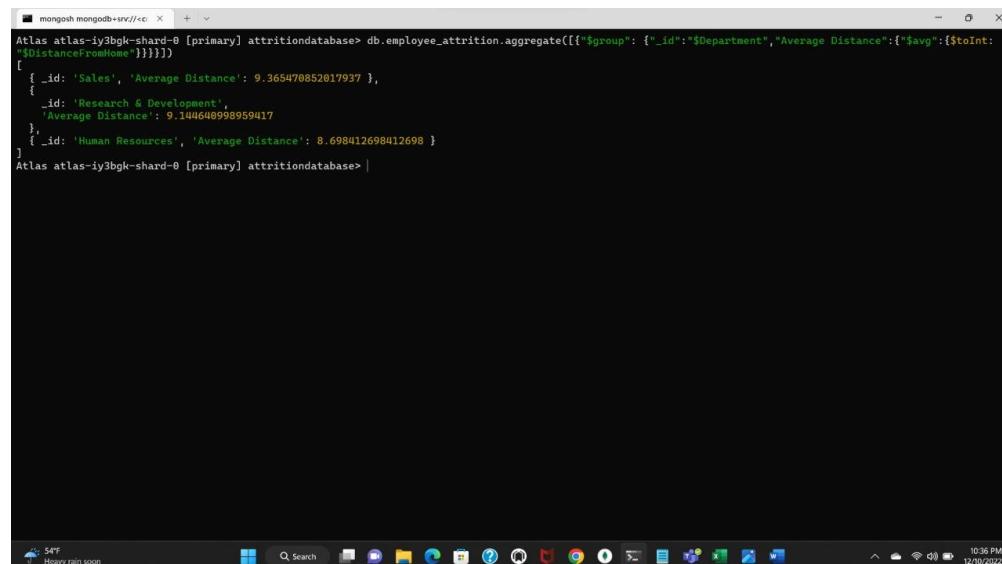
- Result set has 3 documents
- We are using "DistanceFromHome" attribute to calculate the distance between work and home for employees in each department

Translation:

Query the "employee_attrition" collection from the "attritiondatabase". From the collection, use "DistanceFromHome" to calculate average distance for each department and show the distance in result

Screen Shot of MongoDB Query without the count of employees:

CODE: db.employee_attrition.aggregate([{"\$group": {"_id": "\$Department", "Average Distance": {"\$avg": {"\$toInt": "\$DistanceFromHome"} }}}])

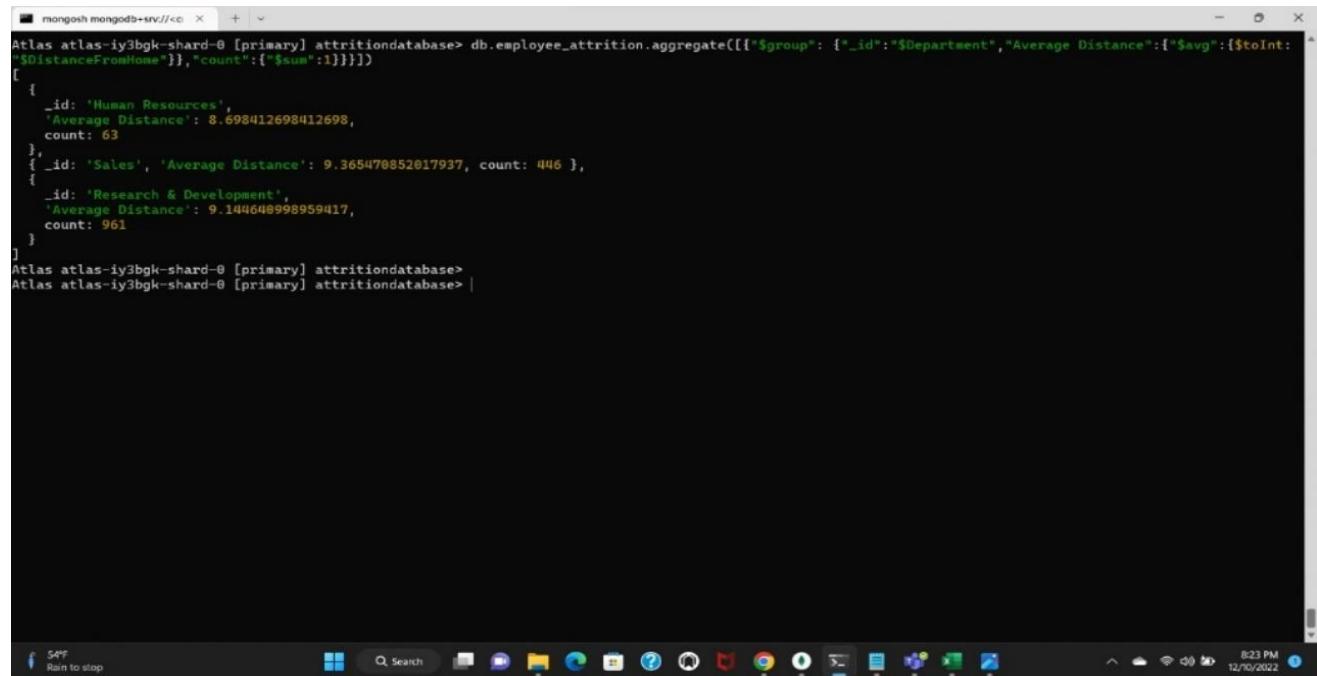


The screenshot shows a terminal window titled "mongosh mongoDB+srv//<local>" running on a Windows operating system. The command entered is "db.employee_attrition.aggregate([{\$group: {_id: '\$Department', 'Average Distance': {\$avg: {\$toInt: '\$DistanceFromHome'}}}}])". The output displays the average distance from home for three departments: Sales (9.3654), Research & Development (9.1446), and Human Resources (8.6984). The terminal window is located at the bottom of the screen, with the taskbar visible below it.

```
mongosh mongoDB+srv//<local>
Atlas atlas-iy3bgk-shard-0 [primary] attritiondatabase> db.employee_attrition.aggregate([{$group: {_id: '$Department', 'Average Distance': {$avg: {$toInt: '$DistanceFromHome'}}}}])
[
  { _id: 'Sales', 'Average Distance': 9.365470852017937 },
  { _id: 'Research & Development',
    'Average Distance': 9.144640998959417 },
  { _id: 'Human Resources', 'Average Distance': 8.698412698412698 }
]
Atlas atlas-iy3bgk-shard-0 [primary] attritiondatabase>
```

Screen Shot of MongoDB Query with the count of employees

CODE: db.employee_attrition.aggregate([{"\$group": {"_id": "\$Department", "Average Distance": {"\$avg": {\$toInt: "\$DistanceFromHome"}}, "count": {"\$sum": 1}}}])



A screenshot of a terminal window titled "mongosh mongodbservice". The command entered is: db.employee_attrition.aggregate([{"\$group": {"_id": "\$Department", "Average Distance": {"\$avg": {\$toInt: "\$DistanceFromHome"}}, "count": {"\$sum": 1}}}])

The output shows the following data:

```
[{"_id": "Human Resources", "Average Distance": 8.698412698412698, "count": 63}, {"_id": "Sales", "Average Distance": 9.365470852017937, "count": 446}, {"_id": "Research & Development", "Average Distance": 9.144648998959417, "count": 961}]
```

Atlas atlas-iy3bgk-shard-0 [primary] attritiondatabase>

Atlas atlas-iy3bgk-shard-0 [primary] attritiondatabase> |

The terminal window is running on a Windows operating system, as indicated by the taskbar icons at the bottom.

Mongo Query 2

Question 5: An Employee in the Sales department has complained to HR that females are paid less than males in the company, in all the departments. What insight can you provide to prove/disprove the statement?

Approach:

To test the claim, we calculated average monthly income of both male and female from all the departments. we found the following details

Average monthly income of “male” and “female” from “sales” department = 6949.64 & 6972.12

Average monthly income of “male” and “female” from “Research and Development” department = 6129.88 & 6513.69

Average monthly income of “male” and “female” from “Human Resources” department = 6371.82 & 7264

Out of the 3 departments, 2 departments pay high to females compared to males. so, the claim is proved to be false

Translation:

Query the “employee_attrition” collection from the “attritiondatabase”. From the collection, use “MonthlyIncome” to calculate average monthly income for each department and show in result

Notes:

- 1) The result contains 6 documents
- 2) We are using MonthlyIncome attribute to calculate average monthly income of male and female from each department

Screen Shot of MongoDB Query without the count of employees

CODE: db.employee_attrition.aggregate([{\$group:{_id:{Department: '\$Department', Gender: '\$Gender'}, AverageStockOptionLevel:{\$avg:{\$toInt: "\$StockOptionLevel"}},AverageMonthlyIncome:{\$avg:{\$toInt: "\$MonthlyIncome"}}}}]])

```
Atlas atlas-iy3bgk-shard-0 [primary] attritiondatabase> db.employee_attrition.aggregate([{$group:{_id:{Department: '$Department', Gender: '$Gender'}, AverageStockOptionLevel:{$avg:{$toInt: "$StockOptionLevel"}},AverageMonthlyIncome:{$avg:{$toInt: "$MonthlyIncome"}}}}]])  
[  
  {  
    _id: { Department: 'Human Resources', Gender: 'Female' },  
    AverageStockOptionLevel: 0.85,  
    AverageMonthlyIncome: 7264  
  },  
  {  
    _id: { Department: 'Sales', Gender: 'Male' },  
    AverageStockOptionLevel: 0.754863813229572,  
    AverageMonthlyIncome: 6949.645914396887  
  },  
  {  
    _id: { Department: 'Sales', Gender: 'Female' },  
    AverageStockOptionLevel: 0.798941798941799,  
    AverageMonthlyIncome: 6972.126984126984  
  },  
  {  
    _id: { Department: 'Research & Development', Gender: 'Male' },  
    AverageStockOptionLevel: 0.8281786941589757,  
    AverageMonthlyIncome: 6129.888316151203  
  },  
  {  
    _id: { Department: 'Research & Development', Gender: 'Female' },  
    AverageStockOptionLevel: 0.7678180263852242,  
    AverageMonthlyIncome: 6513.691292875989  
  },  
  {  
    _id: { Department: 'Human Resources', Gender: 'Male' },  
    AverageStockOptionLevel: 0.7411860465116279,  
    AverageMonthlyIncome: 6371.023255813953  
  }  
]  
Atlas atlas-iy3bgk-shard-0 [primary] attritiondatabase>
```

Screen Shot of MongoDB Query with the count of employees

CODE:

db.employee_attrition.aggregate([{\$group:{_id:{Department:'\$Department',Gender:'\$Gender'},Average StockOptionLevel:{\$avg:{\$toInt:"\$StockOptionLevel"}},AverageMonthlyIncome:{\$avg:{\$toInt:"\$MonthlyIncome"}}, "count": {"\$sum":1}}}]])

```
mongosh mongodb+srv://<c> + ~  
Atlas atlas-iy3bgk-shard-0 [primary] attritiondatabase> db.employee_attrition.aggregate([{$group:{_id:{Department: '$Department', Gender: '$Gender'}, AverageStockOptionLevel:{$avg:{$toInt: "$StockOptionLevel"}},AverageMonthlyIncome:{$avg:{$toInt: "$MonthlyIncome"}}, "count": {"$sum":1}}}]])  
[  
  {  
    _id: { Department: 'Research & Development', Gender: 'Female' },  
    AverageStockOptionLevel: 0.7678180263852242,  
    AverageMonthlyIncome: 6513.691292875989,  
    count: 379  
  },  
  {  
    _id: { Department: 'Research & Development', Gender: 'Male' },  
    AverageStockOptionLevel: 0.8281786941589757,  
    AverageMonthlyIncome: 6129.888316151203,  
    count: 582  
  },  
  {  
    _id: { Department: 'Human Resources', Gender: 'Female' },  
    AverageStockOptionLevel: 0.85,  
    AverageMonthlyIncome: 7264,  
    count: 20  
  },  
  {  
    _id: { Department: 'Human Resources', Gender: 'Male' },  
    AverageStockOptionLevel: 0.7411860465116279,  
    AverageMonthlyIncome: 6371.023255813953,  
    count: 43  
  },  
  {  
    _id: { Department: 'Sales', Gender: 'Male' },  
    AverageStockOptionLevel: 0.754863813229572,  
    AverageMonthlyIncome: 6949.645914396887,  
    count: 257  
  },  
  {  
    _id: { Department: 'Sales', Gender: 'Female' },  
    AverageStockOptionLevel: 0.798941798941799,  
    AverageMonthlyIncome: 6972.126984126984,  
    count: 189  
  }  
]  
54°F Rain to stop Search < > 11:15 PM 12/10/2022
```

Mongo Query 3

Question 6

A press article in a business magazine has said that at this company, married men have higher performance ratings than divorced or single men. What initial finding can you obtain from the data to help articulate the company's response in this regard?

Approach:

To test the claim in article, we fetched the details of single , divorced and married men across all the three departments. We found the following details:

Department	Single/married/Divorced	Average Rating
Research and Development	Single, Divorced	3.177, 3.127
Research and Development	married	3.172
Sales	Single, Divorced	3.125,3.140
Sales	married	3.107
Human Resources	Single, Divorced	3.25,3.25
Human Resources	married	3.043

Out of 3 departments, only in one department (Research and development) , Married men are having an upper hand than divorced men (although this is a very minute difference)

In the remaining departments, Single or Divorced men are having comparatively upper hand than divorced men (although this is a very minute difference)

Notes/Comments About MongoDB Query/Code and Results (Include # of Documents in Result):

- 1) The result contains 9 documents
- 2) We have observed ratings using “PerformanceRating” attribute in the collection
- 3) We have observed the marital status of the individual using “MaritalStatus” attribute in the collection

Translation:

Query the “employee_attrition” collection from the “attritiondatabase”. From the collection, use “PerformanceRating” to observe performance rating for Single, Divorced, married men across each department and show in result

Screen Shot of MongoDB Query without count of employees

CODE: db.employee_attrition.aggregate([{\$match:{Gender: "Male"}},{\$group:{_id: {Department: '\$Department', Gender: "Male", MaritalStatus: '\$MaritalStatus'}, 'Rating': {\$avg: {\$toInt: '\$PerformanceRating'}}}}]).pretty()

```
mongosh mongodb+srv://<...> + - X
Atlas atlas-iy3bgk-shard-0 [primary] attritiondatabase> db.employee_attrition.aggregate([{$match:{Gender:"Male"}},{$group:{_id:{Department:'$Department',Gender:'Male',MaritalStatus:'$MaritalStatus'},'Rating':{$avg:{'$toInt:$PerformanceRating'}}}}]).pretty()
[
  {
    _id: {
      Department: 'Research & Development',
      Gender: 'Male',
      MaritalStatus: 'Married'
    },
    Rating: 3.172932330827068
  },
  {
    _id: { Department: 'Sales', Gender: 'Male', MaritalStatus: 'Single' },
    Rating: 3.125
  },
  {
    _id: { Department: 'Sales', Gender: 'Male', MaritalStatus: 'Divorced' },
    Rating: 3.1403508771929824
  },
  {
    _id: {
      Department: 'Human Resources',
      Gender: 'Male',
      MaritalStatus: 'Single'
    },
    Rating: 3.25
  },
  {
    _id: {
      Department: 'Research & Development',
      Gender: 'Male',
      MaritalStatus: 'Single'
    },
    Rating: 3.177142857142857
  },
  {
    _id: {
      Department: 'Research & Development',
      Gender: 'Male',
      MaritalStatus: 'Divorced'
    }
  }
]
```

```
{
  _id: {
    Department: 'Research & Development',
    Gender: 'Male',
    MaritalStatus: 'Divorced'
  },
  Rating: 3.127659574468085
},
{
  _id: {
    Department: 'Human Resources',
    Gender: 'Male',
    MaritalStatus: 'Married'
  },
  Rating: 3.0434782608695654
},
{
  _id: { Department: 'Sales', Gender: 'Male', MaritalStatus: 'Married' },
  Rating: 3.107142857142857
},
{
  _id: {
    Department: 'Human Resources',
    Gender: 'Male',
    MaritalStatus: 'Divorced'
  },
  Rating: 3.25
}
]
```

Screen Shot of MongoDB Query with the count of employees:

CODE:

```
db.employee_attrition.aggregate([{$match:{Gender:"Male"}},{$group:{_id:{Department:'$Department',MaritalStatus:'$MaritalStatus',Gender:'Male'},'Rating':{$avg:{'$toInt:$PerformanceRating'}}, "count": {"$sum":1}}}],).pretty()
```

```
mongosh mongodb+srv://<...> + | x
Atlas atlas-iy3bgk-shard-0 [primary] attritiondatabase> db.employee_attrition.aggregate([{$match:{Gender:"Male"}},{$group:{_id:{Department:'$Department',MaritalStatus:'$MaritalStatus',Gender:'Male'},'Rating':{$avg:{'$toInt':'$PerformanceRating'}},"count": {"$sum":1}}}],pretty()
[
  {
    _id: {
      Department: 'Research & Development',
      MaritalStatus: 'Married',
      Gender: 'Male'
    },
    Rating: 3.172932330827068,
    count: 266
  },
  {
    _id: { Department: 'Sales', MaritalStatus: 'Married', Gender: 'Male' },
    Rating: 3.107142857142857,
    count: 112
  },
  {
    _id: {
      Department: 'Research & Development',
      MaritalStatus: 'Single',
      Gender: 'Male'
    },
    Rating: 3.177142857142857,
    count: 175
  },
  {
    _id: {
      Department: 'Human Resources',
      MaritalStatus: 'Divorced',
      Gender: 'Male'
    },
    Rating: 3.25,
    count: 12
  },
  {
    _id: {
      Department: 'Human Resources',
      MaritalStatus: 'Married',
      Gender: 'Male'
    }
  }
]
54°F Rain to stop 11:20 PM 12/10/2022
```

```
mongosh mongodb+srv://<...> + | x
},
{
  _id: {
    Department: 'Human Resources',
    MaritalStatus: 'Married',
    Gender: 'Male'
  },
  Rating: 3.0434782608695654,
  count: 23
},
{
  _id: {
    Department: 'Human Resources',
    MaritalStatus: 'Single',
    Gender: 'Male'
  },
  Rating: 3.25,
  count: 8
},
{
  _id: {
    Department: 'Research & Development',
    MaritalStatus: 'Divorced',
    Gender: 'Male'
  },
  Rating: 3.127659574468085,
  count: 141
},
{
  _id: { Department: 'Sales', MaritalStatus: 'Single', Gender: 'Male' },
  Rating: 3.125,
  count: 88
},
{
  _id: { Department: 'Sales', MaritalStatus: 'Divorced', Gender: 'Male' },
  Rating: 3.1403508771929824,
  count: 57
}
]
Atlas atlas-iy3bgk-shard-0 [primary] attritiondatabase>
54°F Rain to stop 11:21 PM 12/10/2022
```

Mongo Query 4

Question 12

A press article in a business magazine has said that at this company, single women in Sales have worked at the company longer than divorced or married women. What initial finding can you obtain from the data to help articulate the company's response in this regard?

Approach:

To test the claim, we have fetched the details of women who are across different departments along with the average number of years they have worked. We found the following details:

Department	Count of female employees	Single/Married/Divorced	Average Number of Years
Sales	94	Married	8.3404
Sales	30	Divorced	7.7666
Sales	65	Single	5.5692

From the above details, we can see that , In the Sales Department, average number of years at company of Single women are less compared to both Married or Divorced women.

Notes/Comments About MongoDB Query/Code and Results (Include # of Documents in Result):

- 1) The result set contains 9 documents
- 2) We used “YearsAtCompany” attribute to calculate the average number of years spent by a particular category of women in Sales Department
- 3) We used “MaritalStatus” attribute to specify the gender in result set
- 4) We used “Department” attribute to fetch the information about women in “Sales” department

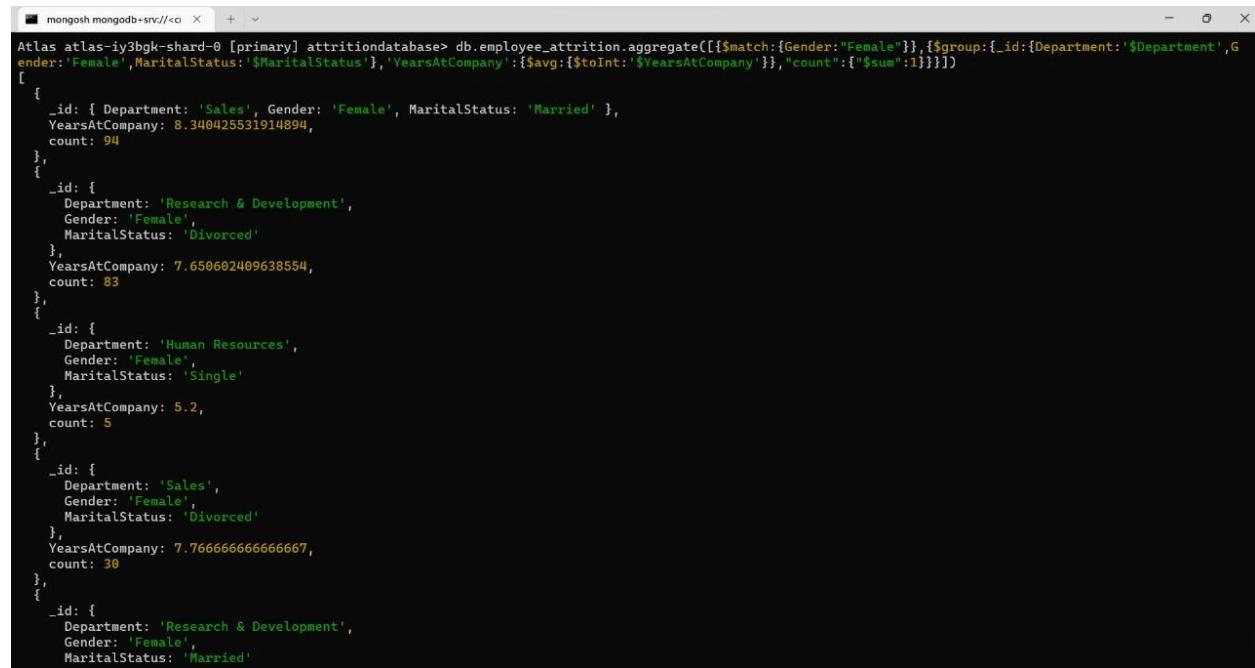
Translation:

Query the “employee_attrition” collection from the “attritiondatabase”. From the collection, use “YearsAtCompany” to calculate the average number of years where the different categories in female such as “Single”, “Divorced”, “Married” has worked at that company across different departments such as “Research and Development”, “Sales”, “Human Resources”

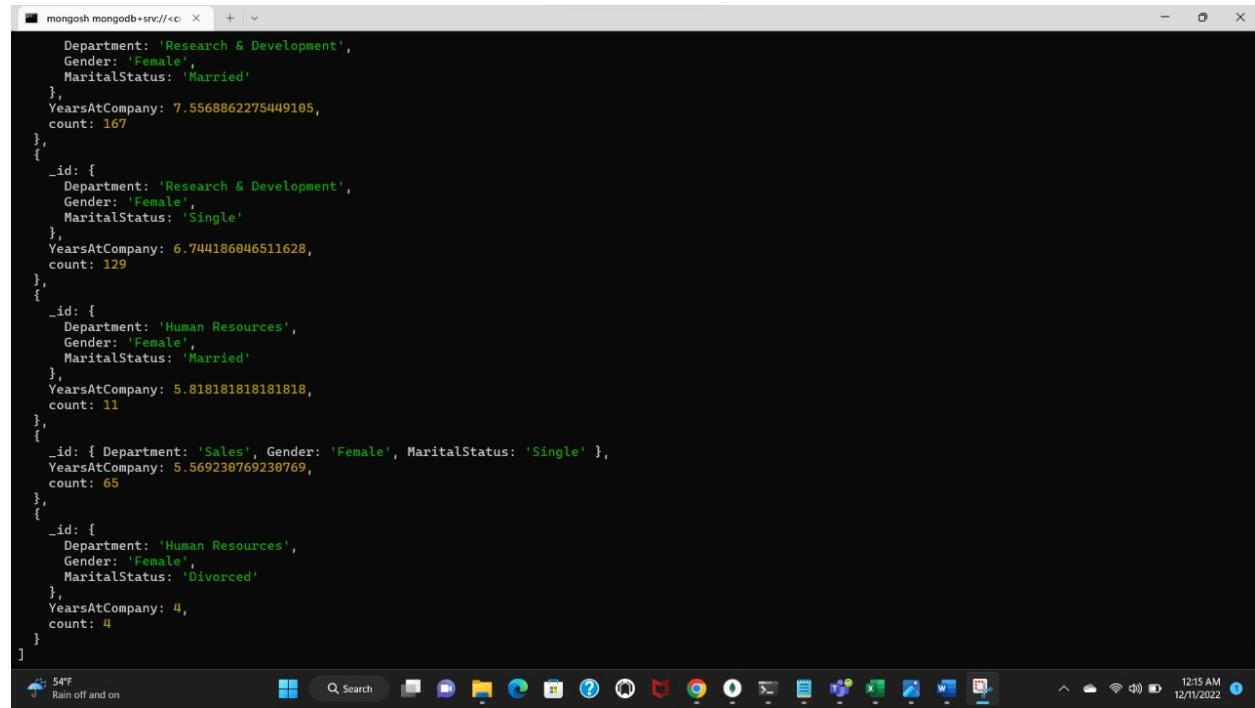
Screen Shot of MongoDB Query with the count of employees:

CODE:

```
db.employee_attrition.aggregate([{$match:{Gender:"Female"}},{$group:{_id:{Department:'$Department',Gender:'Female',MaritalStatus:'$MaritalStatus'},'YearsAtCompany':{$avg:{$toInt:'$YearsAtCompany'}}, "count": {"$sum":1}}}] )
```



```
mongosh mongodb+srv://<...> - + x
Atlas atlas-iy3bgk-shard-0 [primary] attritiondatabase> db.employee_attrition.aggregate([{$match:{Gender:"Female"}},{$group:{_id:{Department:'$Department',Gender:'Female',MaritalStatus:'$MaritalStatus'},'YearsAtCompany':{$avg:{$toInt:'$YearsAtCompany'}}, "count": {"$sum":1}}}] )
[
  {
    _id: { Department: 'Sales', Gender: 'Female', MaritalStatus: 'Married' },
    YearsAtCompany: 8.340425531914894,
    count: 94
  },
  {
    _id: {
      Department: 'Research & Development',
      Gender: 'Female',
      MaritalStatus: 'Divorced'
    },
    YearsAtCompany: 7.650602409638554,
    count: 83
  },
  {
    _id: {
      Department: 'Human Resources',
      Gender: 'Female',
      MaritalStatus: 'Single'
    },
    YearsAtCompany: 5.2,
    count: 5
  },
  {
    _id: {
      Department: 'Sales',
      Gender: 'Female',
      MaritalStatus: 'Divorced'
    },
    YearsAtCompany: 7.766666666666667,
    count: 30
  },
  {
    _id: {
      Department: 'Research & Development',
      Gender: 'Female',
      MaritalStatus: 'Married'
    }
  }
]
```

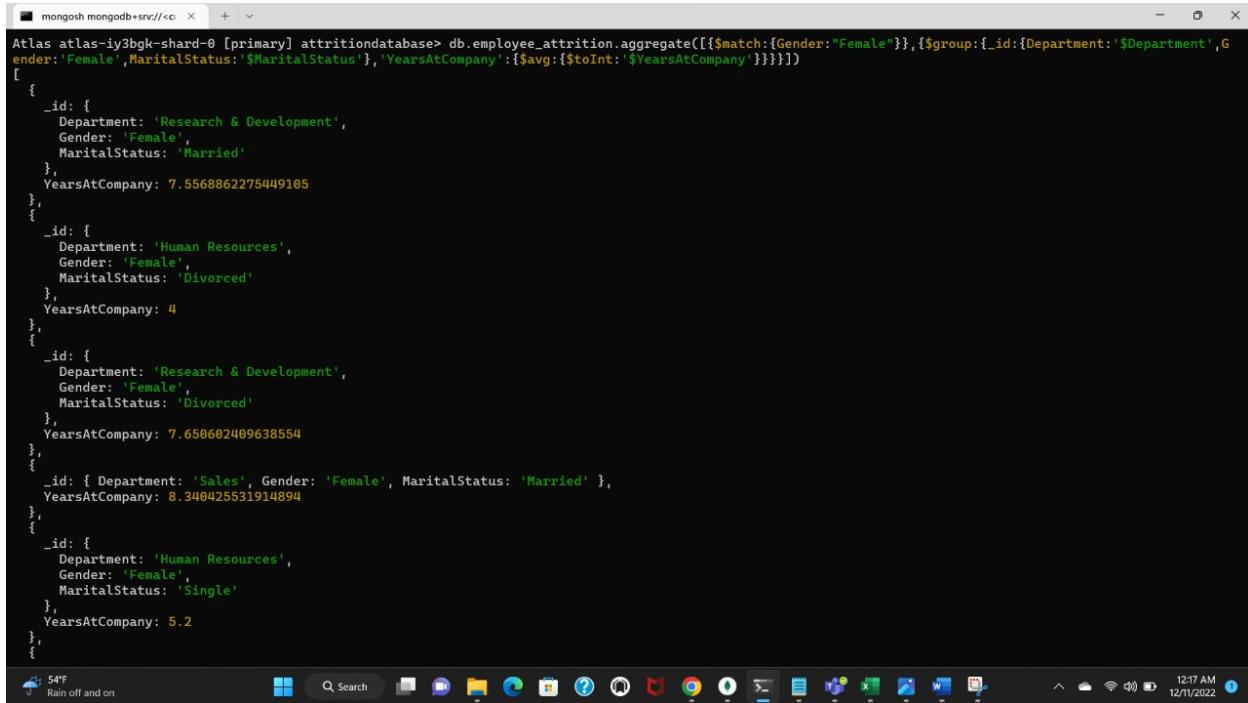


```
mongosh mongodb+srv://<...> - + x
Atlas atlas-iy3bgk-shard-0 [primary] attritiondatabase> db.employee_attrition.aggregate([{$match:{Gender:"Female"}},{$group:{_id:{Department:'$Department',Gender:'Female',MaritalStatus:'$MaritalStatus'},'YearsAtCompany':{$avg:{$toInt:'$YearsAtCompany'}}, "count": {"$sum":1}}}] )
[
  {
    _id: {
      Department: 'Research & Development',
      Gender: 'Female',
      MaritalStatus: 'Married'
    },
    YearsAtCompany: 7.5568862275449105,
    count: 167
  },
  {
    _id: {
      Department: 'Research & Development',
      Gender: 'Female',
      MaritalStatus: 'Single'
    },
    YearsAtCompany: 6.744186046511628,
    count: 129
  },
  {
    _id: {
      Department: 'Human Resources',
      Gender: 'Female',
      MaritalStatus: 'Married'
    },
    YearsAtCompany: 5.818181818181818,
    count: 11
  },
  {
    _id: {
      Department: 'Sales', Gender: 'Female', MaritalStatus: 'Single' },
    YearsAtCompany: 5.569230769230769,
    count: 65
  },
  {
    _id: {
      Department: 'Human Resources',
      Gender: 'Female',
      MaritalStatus: 'Divorced'
    },
    YearsAtCompany: 4,
    count: 4
  }
]
```

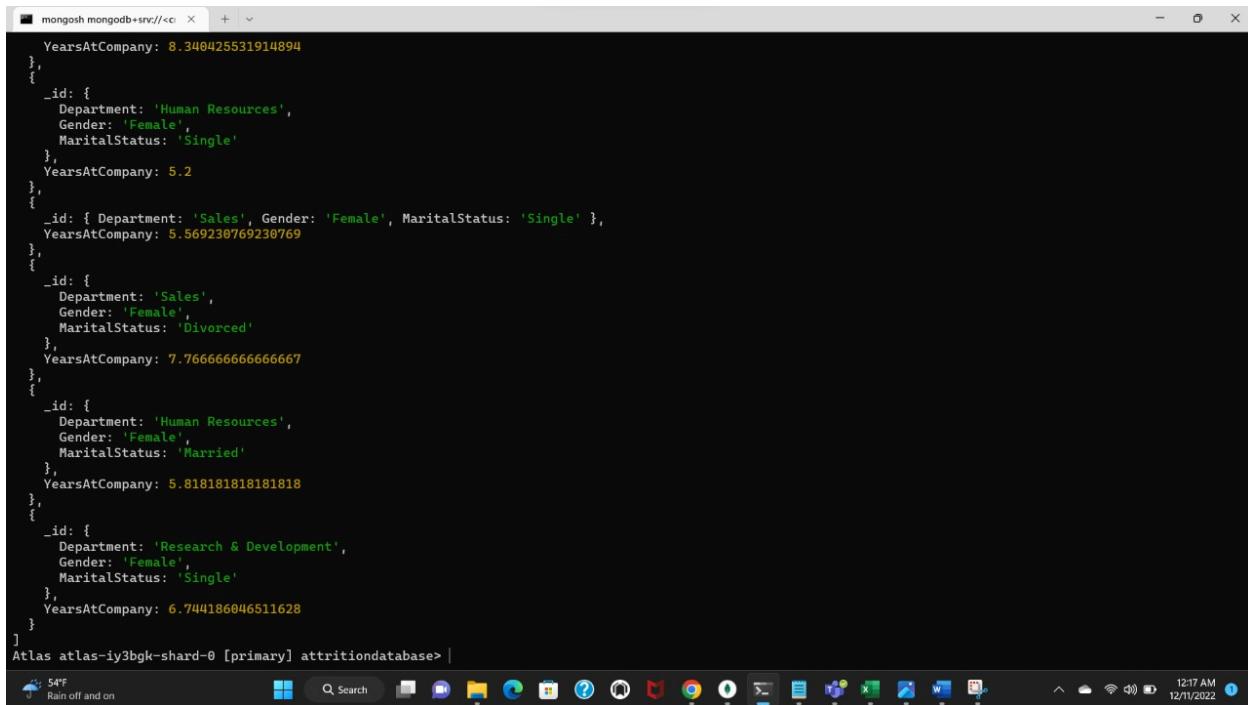
Screen Shot of MongoDB Query without the count of employees:

CODE:

```
db.employee_attrition.aggregate([{$match:{Gender:"Female"}},{$group:{_id:{Department:'$Department',Gender:'Female',MaritalStatus:'$MaritalStatus'},'YearsAtCompany':{$avg:{$toInt:'$YearsAtCompany'}}}}])
```



```
mongosh mongodb+srv://<...> + v
Atlas atlas-iy3bgk-shard-0 [primary] attritiondatabase> db.employee_attrition.aggregate([{$match:{Gender:"Female"}},{$group:{_id:{Department:'$Department',Gender:'Female',MaritalStatus:'$MaritalStatus'},'YearsAtCompany':{$avg:{$toInt:'$YearsAtCompany'}}}}])
[
  {
    _id: {
      Department: 'Research & Development',
      Gender: 'Female',
      MaritalStatus: 'Married'
    },
    YearsAtCompany: 7.5568862275449105
  },
  {
    _id: {
      Department: 'Human Resources',
      Gender: 'Female',
      MaritalStatus: 'Divorced'
    },
    YearsAtCompany: 4
  },
  {
    _id: {
      Department: 'Research & Development',
      Gender: 'Female',
      MaritalStatus: 'Divorced'
    },
    YearsAtCompany: 7.650602409638554
  },
  {
    _id: { Department: 'Sales', Gender: 'Female', MaritalStatus: 'Married' },
    YearsAtCompany: 8.340425531914894
  },
  {
    _id: {
      Department: 'Human Resources',
      Gender: 'Female',
      MaritalStatus: 'Single'
    },
    YearsAtCompany: 5.2
  }
]
mongosh mongodb+srv://<...> + v
12:17 AM 12/11/2022 ①
```



```
mongosh mongodb+srv://<...> + v
YearsAtCompany: 8.340425531914894
},
{
  _id: {
    Department: 'Human Resources',
    Gender: 'Female',
    MaritalStatus: 'Single'
  },
  YearsAtCompany: 5.2
},
{
  _id: { Department: 'Sales', Gender: 'Female', MaritalStatus: 'Single' },
  YearsAtCompany: 5.569230769230769
},
{
  _id: {
    Department: 'Sales',
    Gender: 'Female',
    MaritalStatus: 'Divorced'
  },
  YearsAtCompany: 7.766666666666666
},
{
  _id: {
    Department: 'Human Resources',
    Gender: 'Female',
    MaritalStatus: 'Married'
  },
  YearsAtCompany: 5.818181818181818
},
{
  _id: {
    Department: 'Research & Development',
    Gender: 'Female',
    MaritalStatus: 'Single'
  },
  YearsAtCompany: 6.744186046511628
}
]
Atlas atlas-iy3bgk-shard-0 [primary] attritiondatabase> |
```

