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

Snekha Duppathi

Contents

Relational Data Model

Assumptions/Notes About Data Entities and Relationships

This fictional dataset tells about the employee attrition on a set of population with over 1400 employees. Several factors are laid out to discover the reasons that caused the attrition. Various data entities and relationships among them are drawn up such as the employee's education, their department, salary and several other factors. The subjective factors such as environment satisfaction, job relationship, job satisfaction, performance rating takes a deep dive to provide insights about the reason for attrition. Certain assumptions are laid down and have been taken into consideration on the Entity Relationship Diagram with respect to the data entities and relationships among them. They are as follows:

- Education and Education field are not related. The education column is associated with the level of education an employee has done and the education field is associated with the branch of field the employee studied.

- The columns Job level and Job role are related. Job level represents the skill, workload and expertise associated for each job role with 1 being the lowest and 5 being the highest.

- Age and Over18 are related. If age is greater than or equal to 18 then Over18 is set to 'Y'. So they are placed in separate tables to ensure there are no transitive functional dependencies.

- Daily rate is the amount of money the employee gets paid per day

- Hourly rate is the amount of money paid for the employee per hour

- Monthly rate is the amount of money the employee is paid per month if they work fully in a month without taking leaves and fulfill the employer's expectations

- The monthly income of an employee is based on his education, education field, past work experience, skills he is good at, overtime and so it is different for each employee. The income is not fixed on any specific rates in a generalized manner for everyone.

- From the given data there is no stable relationship between daily rates, hourly rates and monthly rates so they are assumed to be not related with each other or the monthly income of an employee.

- Standard hours are the minimum number of hours according to human resource department that each employee is expected to work.

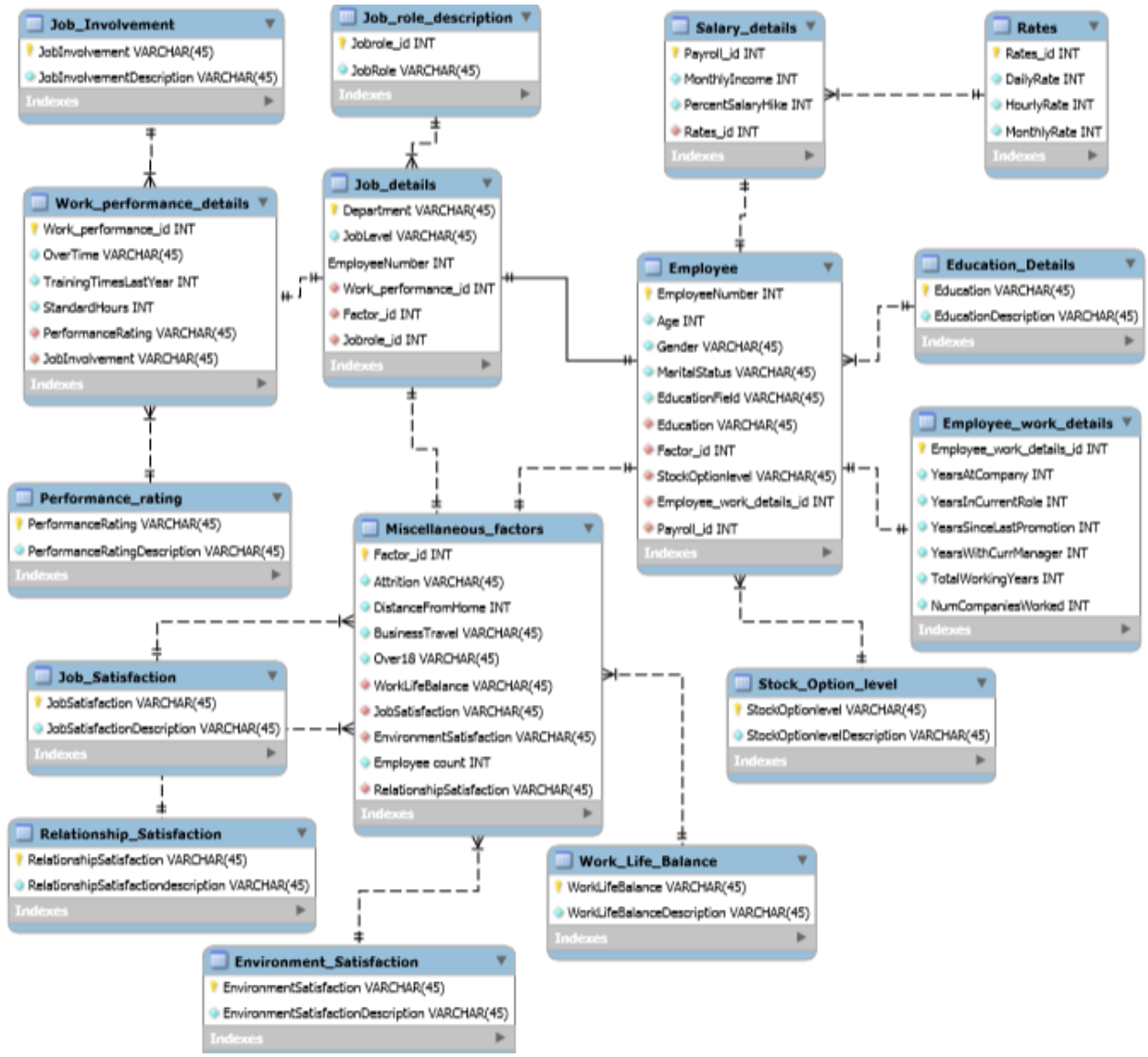
- Employee count is a dummy field to specify that data is collected individually from each employee.

- Overtime is the field that indicates whether the employee has worked after the standard working hours. If he has worked overtime then it will be mentioned as Yes, else No.
- Percentsalaryhike is the percent change in annual salary from 2016 vs 2015.
- Each Jobrole has a unique JobRole_id which is associated with it.
- Each employee has work_performance_id which can be used to retrieve details about a his/her performance. This work performance also depends on the job involvement that he was having in his job.
- The miscellaneous_factors table will have various factors which can give information about the employee's attrition.
- Job satisfaction, Relationship satisfaction, Environment Satisfaction and Work life balance are the few of the factors which let us know about the attrition of an employee. Hence the Miscellaneous_factors table is related with all of them.
- Employee_work_details table will have the information about the employee's past and present career curve.
- Each employee will have a unique Rates_id with different hourly, daily and monthly rates depending upon various factors.
- The description for different stock option levels are newly added; which indicates the stock option that an employee has.
- Few new columns are inserted as primary keys in few tables with newly added data to construct the model in third normalized form.

The data model is in 3NF because:

- Each table has a unique identifier which is a primary key.
- Every primary key has a single value.
- No tables have multivalued or multipart rows in them.
- Each table is linked to other table using a foreign key.
- There are no partial dependencies between the columns in a table.
- There are no transitive dependencies between the columns in a table.

Entity-Relationship Diagram



The model can be found here.

EmployeeAttrition.mwb

Physical MySQL Database

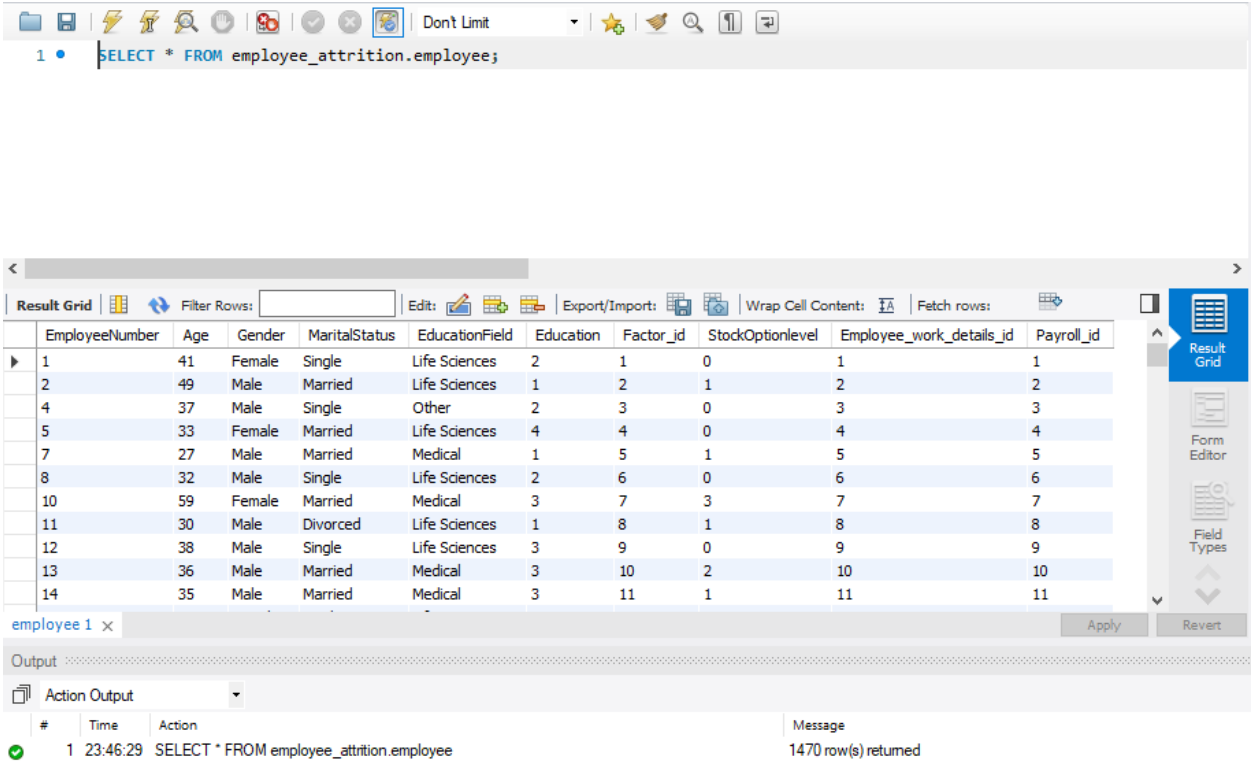
Assumptions/Notes About Data Set

The Employee attrition data is a fictional data set with more than 1400 records of employees classified based on age, gender and various other factors.

Screen shot of Physical Database objects

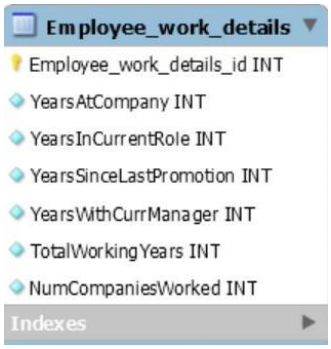
The Employee Attrition Database contains 16 tables

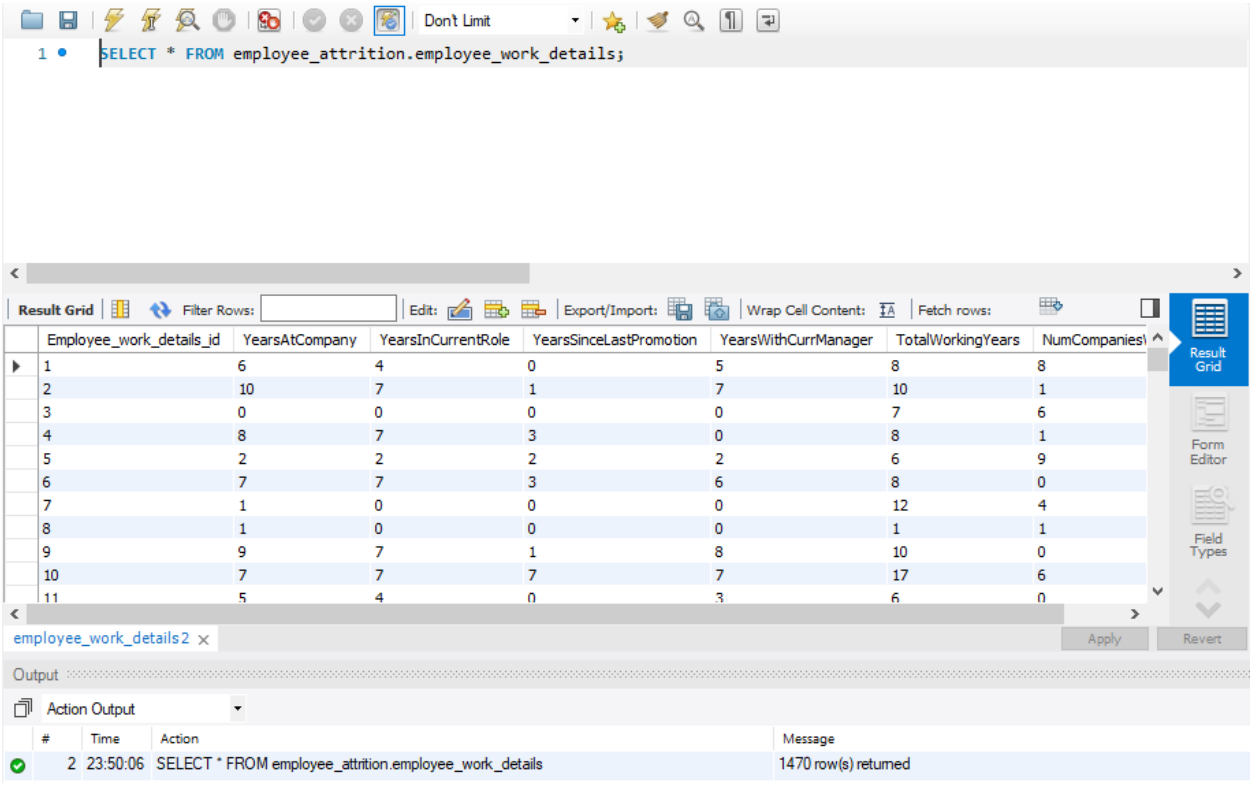
1) Employee table



EmployeeNumber	Age	Gender	MaritalStatus	EducationField	Education	Factor_id	StockOptionlevel	Employee_work_details_id	Payroll_id
1	41	Female	Single	Life Sciences	2	1	0	1	1
2	49	Male	Married	Life Sciences	1	2	1	2	2
4	37	Male	Single	Other	2	3	0	3	3
5	33	Female	Married	Life Sciences	4	4	0	4	4
7	27	Male	Married	Medical	1	5	1	5	5
8	32	Male	Single	Life Sciences	2	6	0	6	6
10	59	Female	Married	Medical	3	7	3	7	7
11	30	Male	Divorced	Life Sciences	1	8	1	8	8
12	38	Male	Single	Life Sciences	3	9	0	9	9
13	36	Male	Married	Medical	3	10	2	10	10
14	35	Male	Married	Medical	3	11	1	11	11

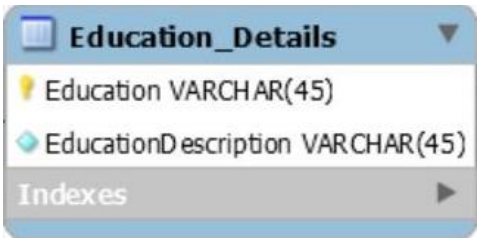
2) Employee_work_details table





Employee_work_details_id	YearsAtCompany	YearsInCurrentRole	YearsSinceLastPromotion	YearsWithCurrManager	TotalWorkingYears	NumCompanies
1	6	4	0	5	8	8
2	10	7	1	7	10	1
3	0	0	0	0	7	6
4	8	7	3	0	8	1
5	2	2	2	2	6	9
6	7	7	3	6	8	0
7	1	0	0	0	12	4
8	1	0	0	0	1	1
9	9	7	1	8	10	0
10	7	7	7	7	17	6
11	5	4	0	3	6	0

3) Education_details table

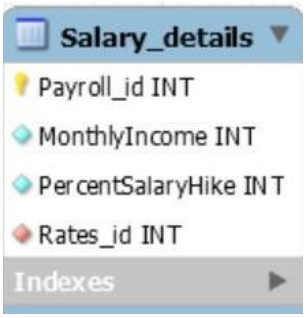


```
SELECT * FROM employee_attrition.education_details;
```

Education	EducationDescription
1	Below College
2	College
3	Bachelor
4	Master
5	Doctor

#	Time	Action	Message
4	23:52:38	SELECT * FROM employee_attrition.education_details	5 row(s) returned

4) Salary_details




```
SELECT * FROM employee_attrition.salary_details;
```

Payroll_id	MonthlyIncome	PercentSalaryHike	Rates_id
1	5993	11	1
2	5130	23	2
3	2090	15	3
4	2909	11	4
5	3468	12	5
6	3068	13	6
7	2670	20	7
8	2693	22	8
9	9526	21	9
10	5237	13	10
11	2426	13	11
12	4193	12	12

#	Time	Action	Message
6	23:55:18	SELECT * FROM employee_attrition.salary_details	1470 row(s) returned

5) Rates table


```
SELECT * FROM employee_attrition.rates;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: | Fetch rows:

Rates_id	DailyRate	HourlyRate	MonthlyRate
1	1102	94	19479
2	279	61	24907
3	1373	92	2396
4	1392	56	23159
5	591	40	16632
6	1005	79	11864
7	1324	81	9964
8	1358	67	13335
9	216	44	8787
10	1299	94	16577
11	809	84	16479
12	153	49	12682

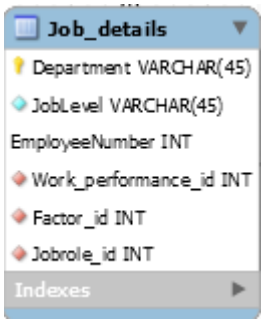
rates 1 | Apply | Revert

Output

Action Output

#	Time	Action	Message
8	23:56:34	SELECT * FROM employee_attrition.rates	1470 row(s) returned

6) Job_details table



```
SELECT * FROM employee_attrition.job_details;
```

Department	JobLevel	EmployeeNumber	Work_performance_id	Factor_id	JobRole_id
Sales	2	1	1	1	1
Research & Development	2	2	2	2	2
Research & Development	1	4	3	3	3
Research & Development	1	5	4	4	2
Research & Development	1	7	5	5	3
Research & Development	1	8	6	6	3
Research & Development	1	10	7	7	3

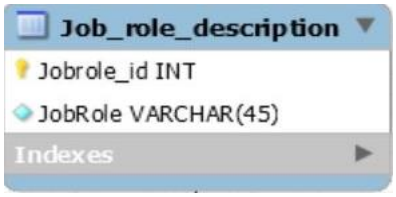
job_details 6

Output

Action Output

#	Time	Action	Message
16	15:57:36	SELECT * FROM employee_attrition.job_details	1470 row(s) returned

7) Job_role_description




```
SELECT * FROM employee_attrition.job_role_description;
```

JobRole_id	JobRole
1	Sales Executive
2	Research Scientist
3	Laboratory Technician
4	Manufacturing Director
5	Healthcare Representative
6	Manager
7	Sales Representative
8	Research Director
9	Human Resources

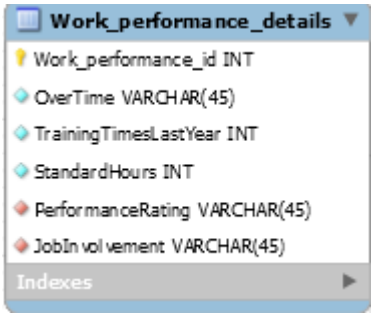
_description 1

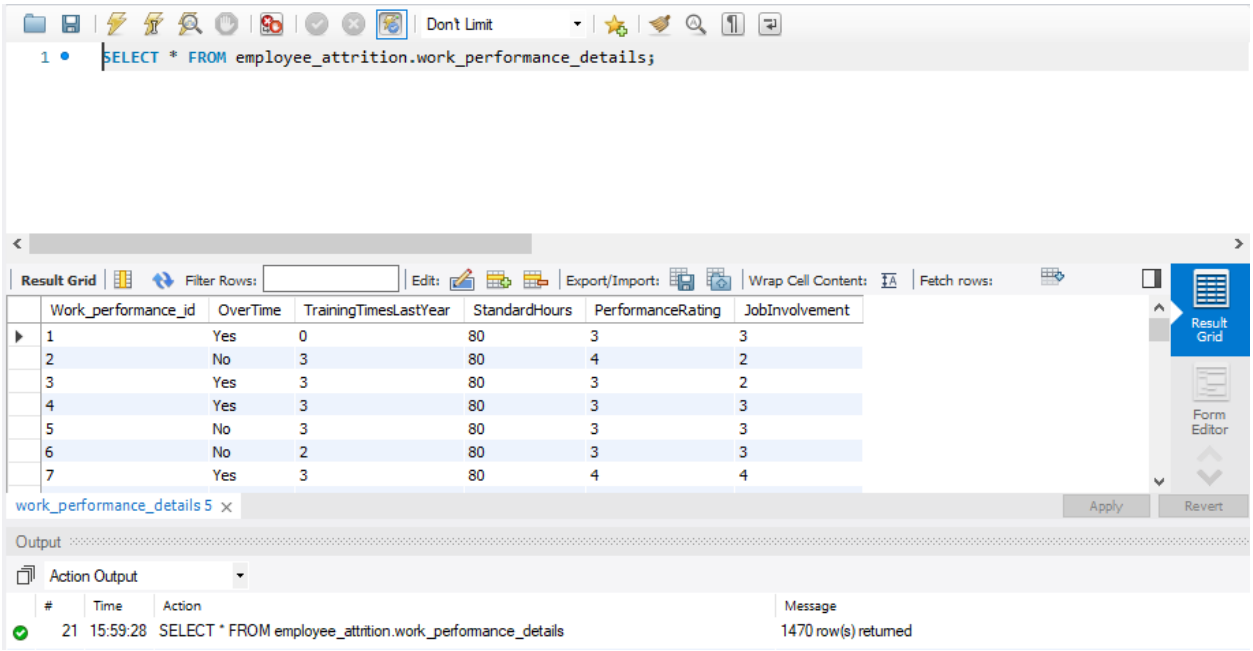
Output

Action Output

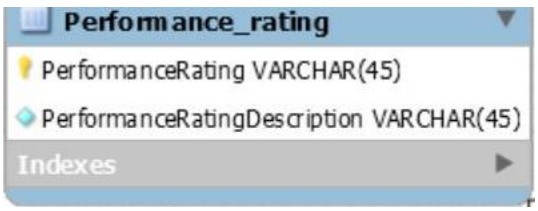
#	Time	Action	Message
12	23:59:38	SELECT * FROM employee_attrition.job_role_description	9 row(s) returned

8) Work_performance_details table





9) Performance_rating table



```
1 •  SELECT * FROM employee_attrition.performance_rating;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

PerformanceRating	PerformanceRatingDescription
3	Excellent
4	Outstanding
NULL	NULL

Result Grid | Form Editor | Field Types

nance_rating 1 ×

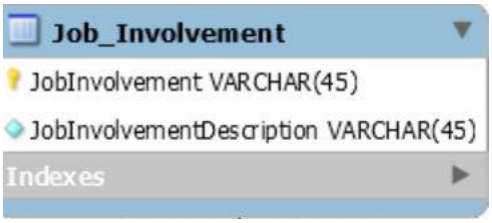
Apply | Revert

Output

Action Output

#	Time	Action	Message
16	00:02:14	SELECT * FROM employee_attrition.performance_rating	2 row(s) returned

10) Job_Involvement table



Don't Limit

```
1 •  SELECT * FROM employee_attrition.job_involvement;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

JobInvolvement	JobInvolvementDescription
1	Low
2	Medium
3	High
4	Very High
NULL	NULL

Result Grid | Form Editor

job_involvement 7 ×

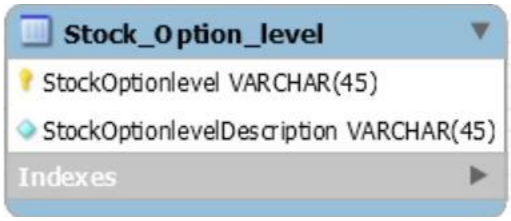
Apply | Revert

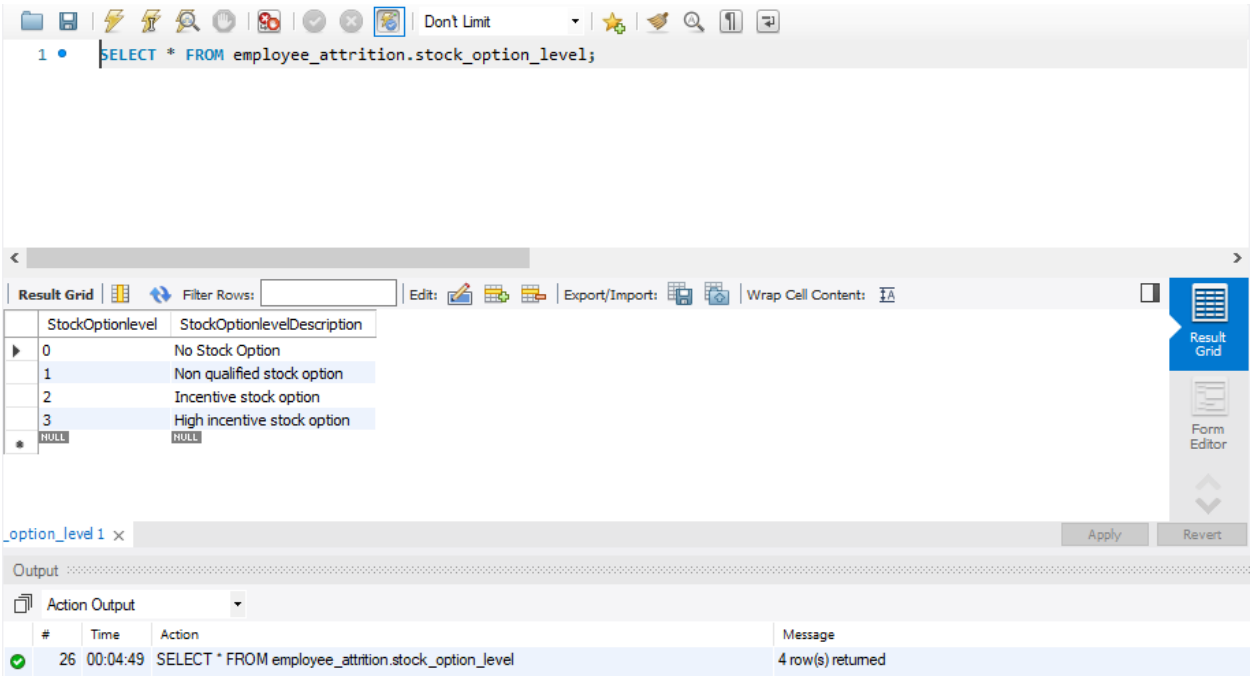
Output

Action Output

#	Time	Action	Message
20	00:03:34	SELECT * FROM employee_attrition.job_involvement	4 row(s) returned

11) Stock_option_level table





12) Miscellaneous_factors table

```
SELECT * FROM employee_attrition.miscellaneous_factors;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: | Fetch rows:

Factor_id	Attrition	DistanceFromHome	BusinessTravel	Over18	WorkLifeBalance	JobSatisfaction	EnvironmentSatisfaction	Employee count	R
1	Yes	1	Travel_Rarely	Y	1	4	2	1	1
2	No	8	Travel_Frequently	Y	3	2	3	1	4
3	Yes	2	Travel_Rarely	Y	3	3	4	1	2
4	No	3	Travel_Frequently	Y	3	3	4	1	3
5	No	2	Travel_Rarely	Y	3	2	1	1	4
6	No	2	Travel_Frequently	Y	2	4	4	1	3

eous_factors 1 | Apply | Revert

Output

Action Output

#	Time	Action	Message
32	00:06:16	SELECT * FROM employee_attrition.miscellaneous_factors	1470 row(s) returned

13) Relationship_Satisfaction table


```
SELECT * FROM employee_attrition.relationship_satisfaction;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

RelationshipSatisfaction	RelationshipSatisfactiondescription
1	Low
2	Medium
3	High
4	Very High
NULL	NULL

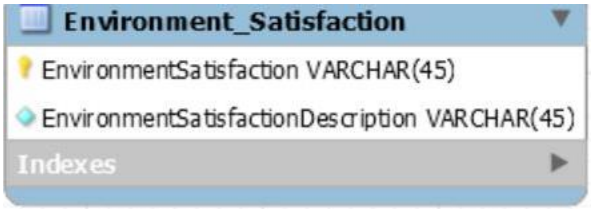
_satisfaction 1 | Apply | Revert

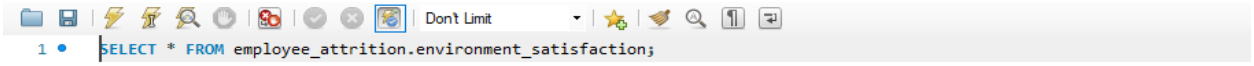
Output

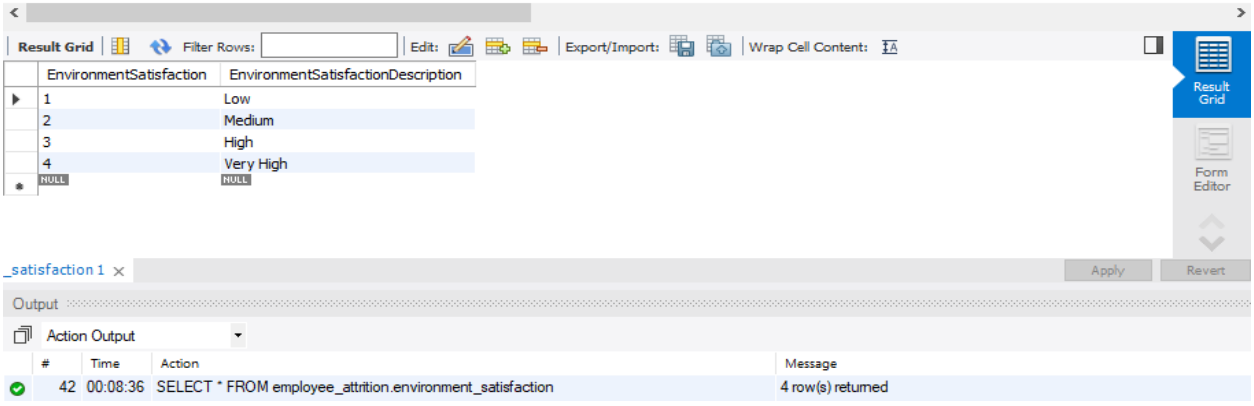
Action Output

#	Time	Action	Message
37	00:07:38	SELECT * FROM employee_attrition.relationship_satisfaction	4 row(s) returned

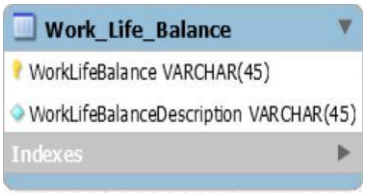
14) Environment_Satisfaction table







15) Work_Life_Balance table



```
1 •  SELECT * FROM employee_attrition.work_life_balance;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

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

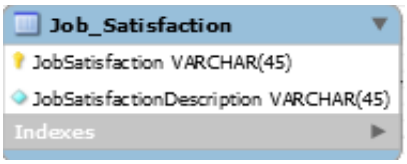
work_life_balance 5 | work_life_balance 6 | work_life_balance 7 | Apply | Revert

Output

Action Output

#	Time	Action	Message
49	00:09:31	SELECT * FROM employee_attrition.work_life_balance	4 row(s) returned

16) Job_Satisfaction table



Don't Limit

```
1 •  SELECT * FROM employee_attrition.job_satisfaction;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

JobSatisfaction	JobSatisfactionDescription
1	Low
2	Medium
3	High
4	Very High
NULL	NULL

job_satisfaction3 | Apply | Revert

Output

Action Output

#	Time	Action	Message
54	00:16:13	SELECT * FROM employee_attrition.job_satisfaction	4 row(s) returned

Data in the Database:

Table Name	Primary Key	Foreign Key	# of Rows in Table
Employee	EmployeeNumber	Education Factor_id StockOptionlevel Employee_work_details_id Payroll_id	1470
Education_Details	Education		5
Employee_work_details	Employee_work_details_id		1470
Job_details	Department EmployeeNumber	Work_performance_id Factor_id Jobrole_id	1470
Job_role_description	Jobrole_id		9
Salary_details	Payroll_id	Rates_id	1470
Rates	Rates_id		1470
Work_performance_details	Work_performance_id	PerformanceRating JobInvolvement	1470
Performance_rating	PerformanceRating		2
Job Involvement	JobInvolvement		4
Stock_Option_Level	StockOptionlevel		4
Miscellaneous_factors	Factor_id	WorkLifeBalance JobSatisfaction EnvironmentSatisfaction RelationshipSatisfaction	1470
Job_Satisfaction	JobSatisfaction		4
Relationship_Satisfaction	RelationshipSatisfaction		4
Environment_Satisfaction	EnvironmentSatisfaction		4
Work_Life_Balance	WorkLifeBalance		4

SQL Queries

SQL Query 1

Question

The company has decided to focus on cost cutting to improve their bottom line in a difficult economy. Should the company focus on Business Travel? Why or Why not?

Notes/Comments About SQL Query and Results (Include # of Rows in Result)

- In this query we have obtained the Business travel frequencies of employees. Out of 1470 people, we got 1043 people who 'Travel_Rarely', 277 people who 'Travel_frequently' and 150 people with 'Non-Travel'.

- Looking at the result we can see that nearly 70% of people travel rarely and almost 19% people travel frequently. Whereas there are around 10% of the people who do not travel at all. **As almost 90% of people are invloved in travelling for business, we can conclude that the cost cutting measure of company on Business travel will have an effect on improving the bottom line in difficult economy.**

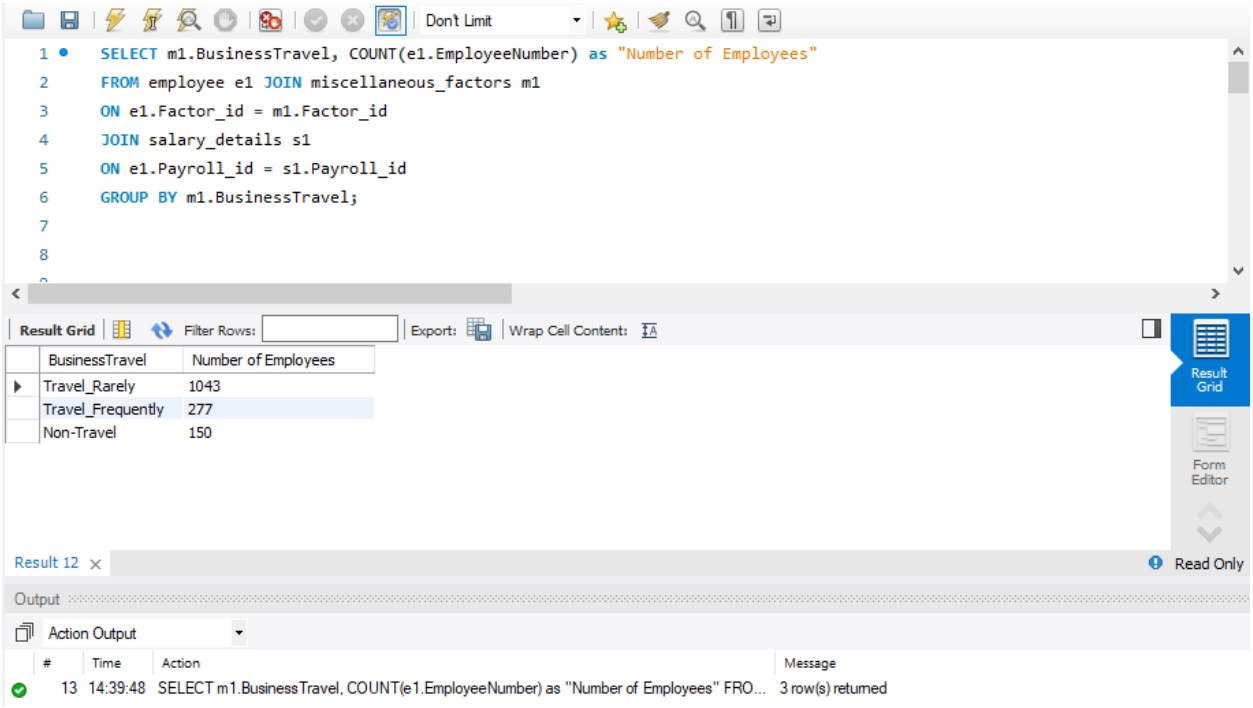
Translation

Translation: Select Business travel type, the count of employees from Employee table joined with Miscellaneous factors table on Factor id joined with Salary details table on Payroll id, grouped by Business travel type

Cleanup: Select BusinessTravel, count(EmployeeNumber) from Employee inner join miscelllaneous_factors on employee.Factor_id = miscellaneous_factors.Factor_id inner join salary_details on employee.Payroll_id = salary_details.Payroll_id group by miscellaneous_factors.BusinessTravel

Code: SELECT m1.BusinessTravel, COUNT(e1.EmployeeNumber) as "Number of Employees" FROM employee e1 JOIN miscellaneous_factors m1 ON e1.Factor_id = m1.Factor_id JOIN salary_details s1 ON e1.Payroll_id = s1.Payroll_id GROUP BY m1.BusinessTravel;

Screen Shot of SQL Query and Results



SQL Query 2

Question

Which department's employee is the most likely to have the longest commute between home and work?

Notes/Comments About SQL Query and Results (Include # of Rows in Result)

- **The employee of 'Sales' and 'Research & Development' department have to travel 29 units from home to work.** This distance is the maximum commute distance for an employee from the respective departments.

- As we were being asked to calculate the longest commute of an employee between home and work, we have used Max function in SQL to find out what is the maximum distance for a specific department, separately.

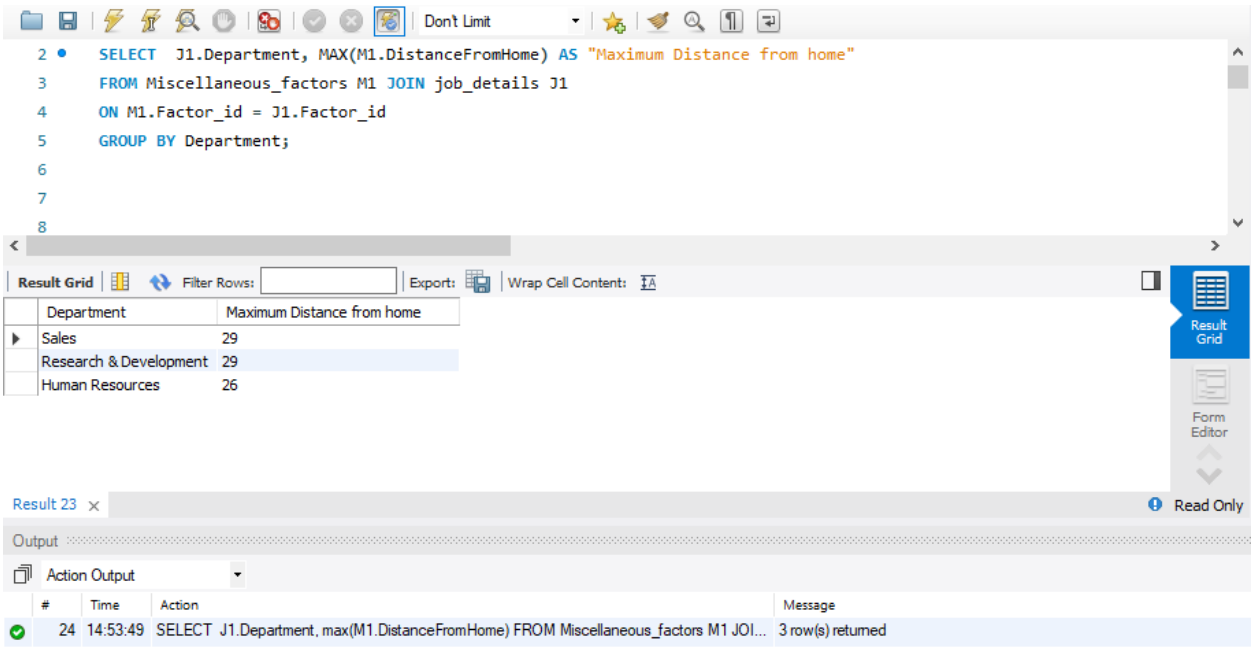
Translation

Translation: Select department, the highest distance from home from Miscellaneous factors table joined with Job details table on Factor id, grouped by department

Cleanup: Select department,Max(Distance from home) from miscellaneous_factors inner join job_details on miscellaneoud_factors. Factor_id = job_details.Factor_id group by department

Code: SELECT J1.Department, MAX(M1.DistanceFromHome) AS "Maximum Distance from home" FROM Miscellaneous_factors M1 JOIN job_details J1 ON M1.Factor_id = J1.Factor_id GROUP BY Department;

Screen Shot of SQL Query and Results



SQL Query 3

Question

A new employee with a Technical Degree 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?

Notes/Comments About SQL Query and Results (Include # of Rows in Result)

- For this query we have obtained that there are 34 people that are having 'Technical Degree' and are working in 'Sales' department.
- Out of total 446 people who work in Sales department the percentage of people having 'Technical Degree' is just around 7.6%. **From this information we can figure out that there are very slim chances for a new employee to join in 'Sales' department with a 'Technical Degree' background.**

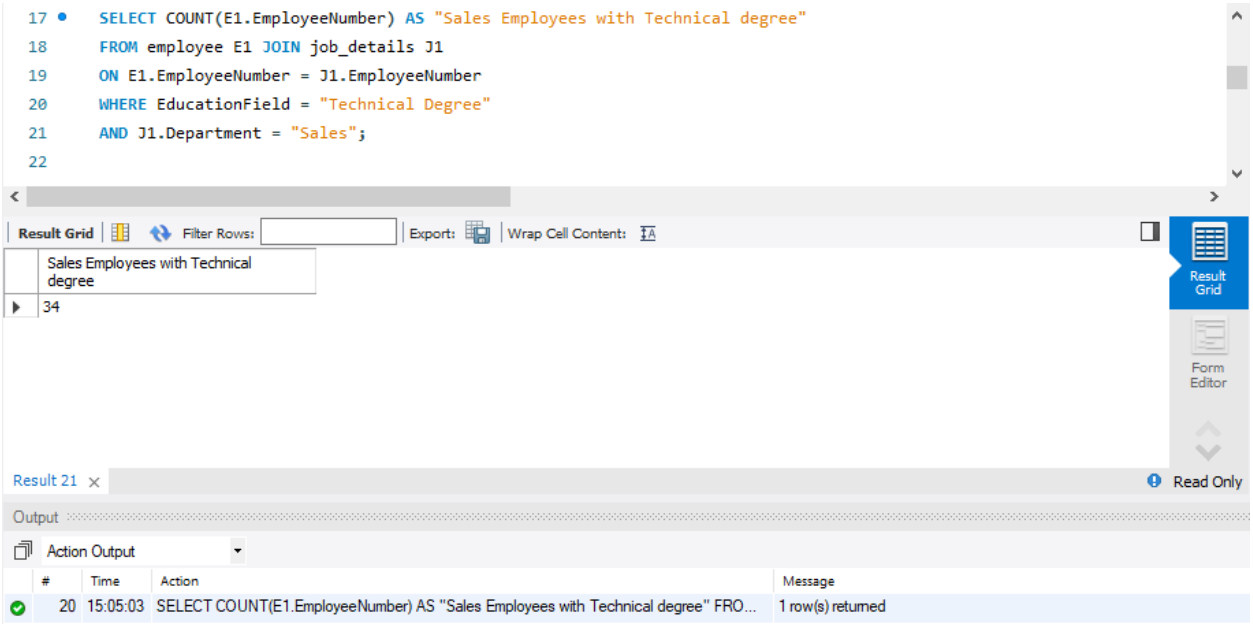
Translation

Translation: Select count of employees from Employee table joined with Job details table on employee number who have an education field as technical degree and belong to Sales department

Cleanup: Select Count(EmployeeNumber) from employee inner join job_details on Employee.EmployeeNumber = job_details.EmployeeNumber where EducationField = "Technical degree" and department="Sales"

Code: SELECT COUNT(E1.EmployeeNumber) AS "Sales Employees with Technical degree" FROM employee E1 JOIN job_details J1 ON E1.EmployeeNumber = J1.EmployeeNumber WHERE EducationField = "Technical Degree" AND J1.Department = "Sales";

Screen Shot of SQL Query and Results



SQL Query 4

Question

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

Notes/Comments About SQL Query and Results (Include # of Rows in Result)

- Running the query provided below we obtained the average of job satisfaction and environment satisfaction of employees for each department.

- The highest job satisfaction was found to be 2.751 for the employees of 'Sales' department. Whereas 'Research & Development' department was having the highest environment satisfaction among the employees. **The claims made in the question by both the departments are found to be right.**

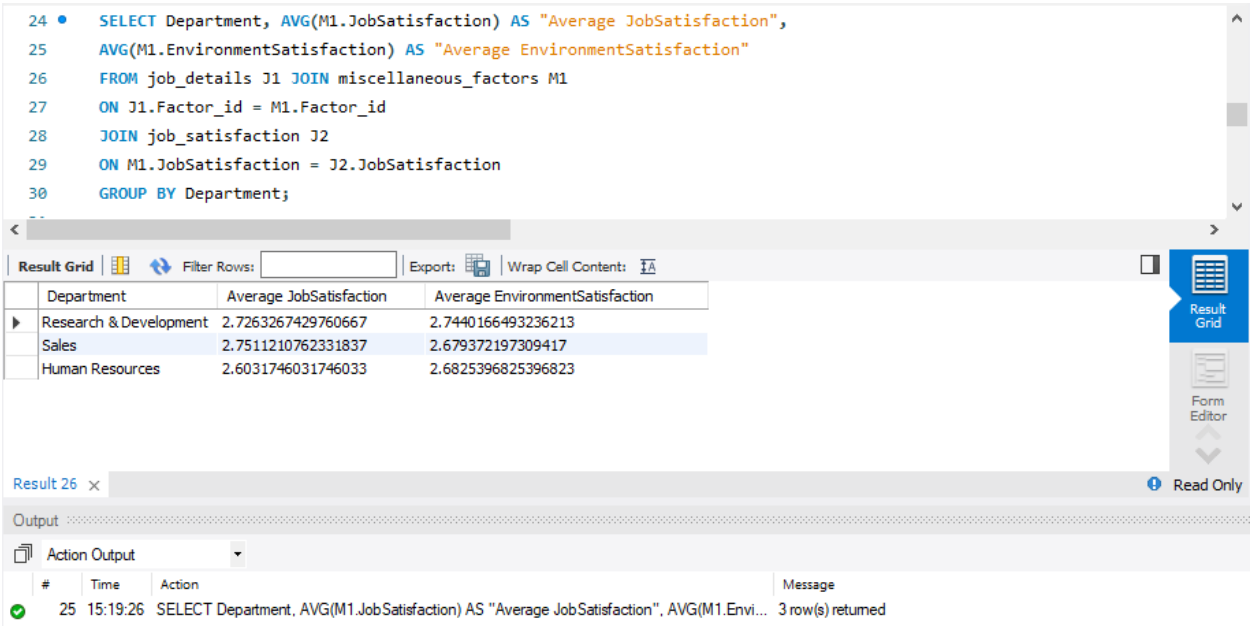
Translation

Translation: Select Department, average of job satisfaction, average of environment satisfaction from job details table joined with miscellaneous factors table on Factor id joined with job satisfaction table on job satisfaction, grouped by department

Cleanup: Select department, avg(job satisfaction), avg(environment satisfaction) from job_details inner join miscellaneous_factors on job_details.Factor_id = miscellaneous_factors.Factor_id inner join job_satisfaction on miscellaneous_factors.JobSatisfaction = job_satisfaction.JobSatisfaction group by department

Code: SELECT Department, AVG(M1.JobSatisfaction) AS "Average JobSatisfaction", AVG(M1.EnvironmentSatisfaction) AS "Average EnvironmentSatisfaction" FROM job_details J1 JOIN miscellaneous_factors M1 ON J1.Factor_id = M1.Factor_id JOIN job_satisfaction J2 ON M1.JobSatisfaction = J2.JobSatisfaction GROUP BY Department;

Screen Shot of SQL Query and Results



SQL Query 5

Question

Company has put out a public statement saying that they have no gender gap when it comes to compensation, in all the departments. What insight can you provide to prove or disprove that statement?

Notes/Comments About SQL Query and Results (Include # of Rows in Result)

- For solving this question, we took the average monthly income of males and females for each of the department.

- In 'Human Resources' department the average salary of 20 female was 7264 units, whereas for 43 male it was 6371.02. In 'Research & Development' 379 female were having an average monthly income of 6513.69 whereas for 582 male it was 6129.88. For the Sales department it was found that 189 female were having 6972.12 as average monthly income, whereas for 257 male it was around 6949.64.

- In all the departments we can see that female have higher monthly income. **From this we can disapprove the claim made by company that they have no gender gap when it comes to compensation in all departments.** The average income of female is higher than male by just marginal value, but as the same pattern was found in every department, we can't say that there is no gender gap.

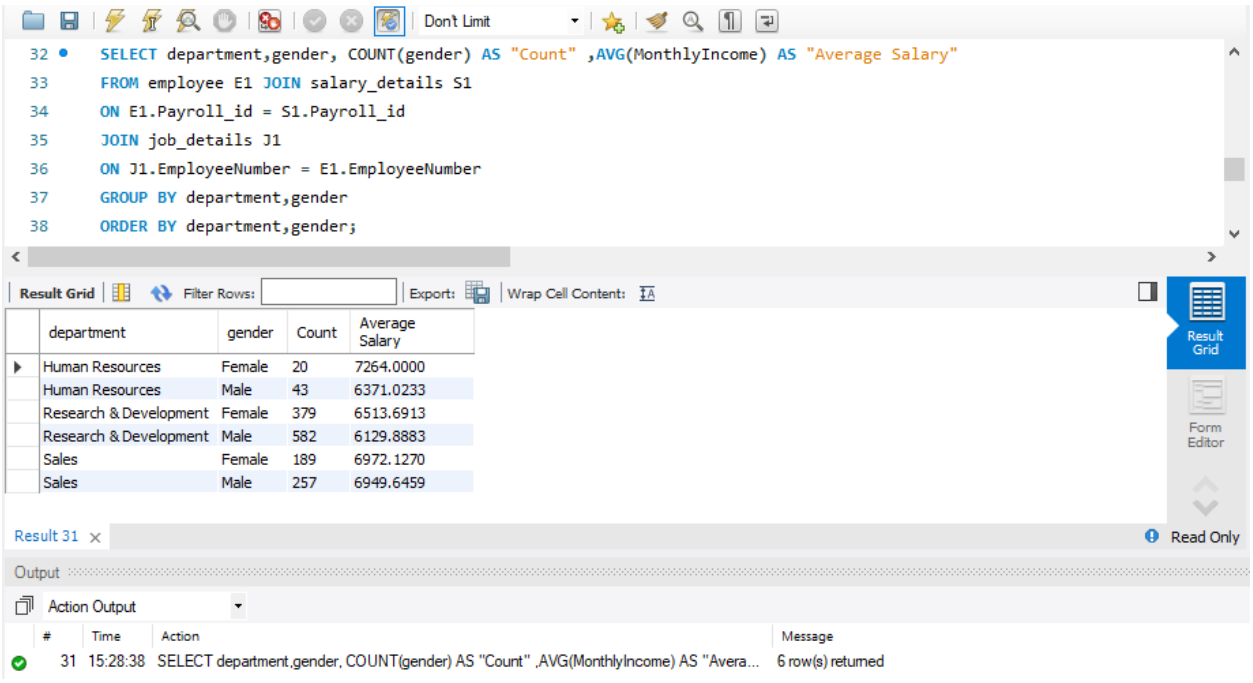
Translation

Translation: Select Department, Gender, count of gender, average of monthly income from employee table joined with salary details table on Payroll id joined with job details table on employee number grouped by department and gender, sorted by department and gender

Cleanup: Select Department, Gender, count(gender),avg(monthly income) from employee inner join salary_details on employee.Payroll_id = salary_details.Payroll_id inner join job_details on job_details.EmployeeNumber = employee.EmployeeNumber group by department, gender order by department, gender

Code: SELECT department, gender, COUNT(gender) AS "Count" ,AVG(MonthlyIncome) AS "Average Salary" FROM employee E1 JOIN salary_details S1 ON E1.Payroll_id = S1.Payroll_id JOIN job_details J1 ON J1.EmployeeNumber = E1.EmployeeNumber GROUP BY department,gender ORDER BY department,gender;

Screen Shot of SQL Query and Results



```
SELECT department,gender, COUNT(gender) AS "Count" ,AVG(MonthlyIncome) AS "Average Salary"
FROM employee E1 JOIN salary_details S1
ON E1.Payroll_id = S1.Payroll_id
JOIN job_details J1
ON J1.EmployeeNumber = E1.EmployeeNumber
GROUP BY department,gender
ORDER BY department,gender;
```

Result Grid

department	gender	Count	Average Salary
Human Resources	Female	20	7264.0000
Human Resources	Male	43	6371.0233
Research & Development	Female	379	6513.6913
Research & Development	Male	582	6129.8883
Sales	Female	189	6972.1270
Sales	Male	257	6949.6459

Result 31 | Read Only

Output

Action Output

#	Time	Action	Message
31	15:28:38	SELECT department,gender, COUNT(gender) AS "Count" ,AVG(MonthlyIncome) AS "Avera...	6 row(s) returned

SQL Query 6

Question

HR is trying to determine whether gender and marital status affect performance ratings of employees in each department. What initial finding can you obtain from the data to help in this regard?

Notes/Comments About SQL Query and Results (Include # of Rows in Result)

- For finding insights for this question we ran a query to check the performance rating of each gender with their marital status of each department. In total 18 rows were obtained to see clear insights.

- **Married female of 'Human Resources' were having best performance rating of 3.27, whereas Single and Divorced female had the worst performance rating average of 3 in the entire group of employees.**

- In 'Human Resources' and 'Research & Development' married female are having better performance rating, in comparison to female of other marital status. While in Sales department divorced female were having best performance rating. Married male are very likely to show poor performance in 'Human Resource' and 'Sales' department in compared to male of other marital status. In 'Research & Development' single male are having better performance in comparison to other males.

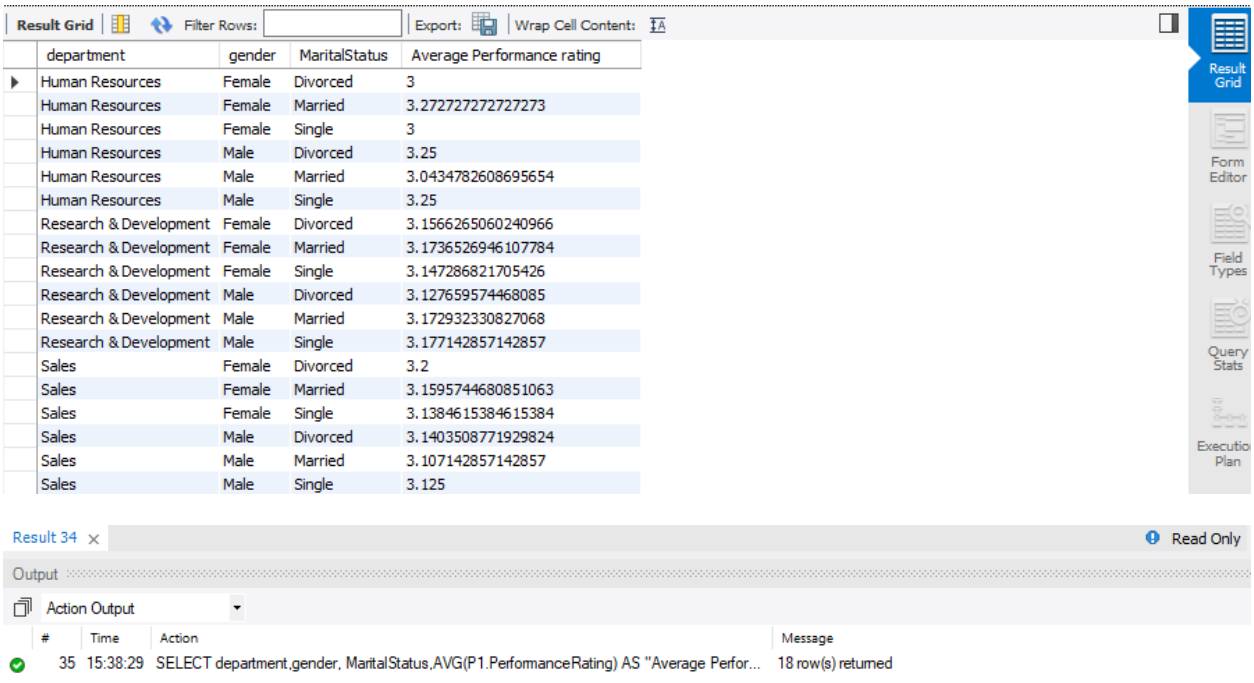
Translation

Translation: Select Department, Gender, marital status, average of performance rating from employee table joined with job details on employee number joined with work performance details table on work performance id again joined with performance rating table on performance rating, grouped by department, gender, marital status and sorted by department

Cleanup: Select Department, Gender, MaritalStatus, avg(PerformanceRating) from employee inner join job_details on employee.EmployeeNumber= job_details.EmployeeNumber inner join work_performance_details on work_performance_details.work_performance_id = job_details. work_performance_id inner join performance_rating on performance_rating.performance_rating = work_performance_details.performance_rating group by department, gender, marital status and order by department

Code: SELECT department, gender, MaritalStatus, AVG(P1.PerformanceRating) AS "Average Performance rating" FROM Employee E1 JOIN job_details J1 ON E1.EmployeeNumber = J1.EmployeeNumber JOIN work_performance_details W1 ON W1.Work_performance_id = J1.Work_performance_id JOIN performance_rating P1 ON P1.PerformanceRating = W1.PerformanceRating GROUP BY Department,gender,MaritalStatus ORDER BY Department;

Screen Shot of SQL Query and Results



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

department	gender	MaritalStatus	Average Performance rating
Human Resources	Female	Divorced	3
Human Resources	Female	Married	3.272727272727273
Human Resources	Female	Single	3
Human Resources	Male	Divorced	3.25
Human Resources	Male	Married	3.0434782608695654
Human Resources	Male	Single	3.25
Research & Development	Female	Divorced	3.1566265060240966
Research & Development	Female	Married	3.1736526946107784
Research & Development	Female	Single	3.147286821705426
Research & Development	Male	Divorced	3.127659574468085
Research & Development	Male	Married	3.172932330827068
Research & Development	Male	Single	3.177142857142857
Sales	Female	Divorced	3.2
Sales	Female	Married	3.1595744680851063
Sales	Female	Single	3.1384615384615384
Sales	Male	Divorced	3.1403508771929824
Sales	Male	Married	3.107142857142857
Sales	Male	Single	3.125

Result Grid | Form Editor | Field Types | Query Stats | Execution Plan

Result 34 | Read Only

Output

Action Output

#	Time	Action	Message
35	15:38:29	SELECT department,gender, MaritalStatus,AVG(P1.PerformanceRating) AS "Average Perfor...	18 row(s) returned

Data Review for MongoDB

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

- We have loaded the entire raw data (CSV file) into the Mongo server using Command prompt and visualized it using Mongo Shell.
- We created a database “employeeattrition” with single collection “attrition” containing the entire data.
- The Mongo queries are written in Mongo Shell which is an interactive JavaScript interface to MongoDB.
- We have verified the results of Mongo DB queries with the SQL query results and it is matching.
- Since MongoDB is document based and hence does not require any normalization and so no row structure is enforced. But in general Relationships in Mongo can be modeled via ‘Embedded’ and ‘Referenced’ approaches. Such relationships can be either 1:1, 1: N, N:1 or N: N

Physical Mongo Database

Assumptions/Notes About Data Set

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

The below screenshot of the physical database is taken in Mongo Shell

```
> use employeeattrition
switched to db employeeattrition
> show collections
attrition
> db.attrition.find().count()
1470
```

Data in the Database

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

MongoDB Queries/Code

Mongo Query 1

Question

Which department's employee is the most likely to have the longest commute between home and work?

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

- We use the aggregate framework to group the employees by department and find the maximum distance from home to work in each department.
- From the results, we can see that the distance from **home to work for the employees in Sales and Research & Development department are the highest with value of 29 each** and so the employees in these 2 departments will have the longest commute between home and work.

Translation

Using attrition collection, utilize aggregate function to group documents by Department (using Department field) and get the maximum distance from home to work (using DistanceFromHome field) of employees in each department

Screen Shot of MongoDB Query/Code and Results

Code:

```
db.attrition.aggregate([{"$group":{_id:'$Department',Distance:{$max:'$DistanceFromHome'}}},{$sort:{Distance:-1}}])
```

```
> db.attrition.aggregate([{"$group":{_id:'$Department',Distance:{$max:'$DistanceFromHome'}}},{$sort:{Distance:
-1}}])
{ "_id" : "Research & Development", "Distance" : 29 }
{ "_id" : "Sales", "Distance" : 29 }
{ "_id" : "Human Resources", "Distance" : 26 }
```

Mongo Query 2

Question

A new employee with a Technical Degree 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?

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

- We assume that from the existing data we can predict the chances of the new employee with technical degree to work in the Sales department
- We use aggregate framework to group the employees based on department to count the number of employees in Sales having a technical degree.
- From the results, we can see that the number of employees who have worked in sales department having a technical degree are 34. Out of total of 446 employees in sales department, only 34 people have a technical degree which is a very proportion of 7%. So, we can conclude that **there is only a marginal and high unlikely chance for the new employee with technical degree to work in Sales department.**

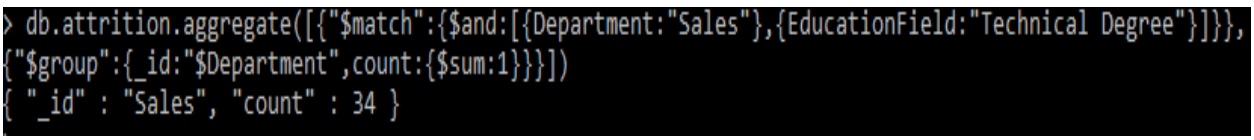
Translation

Using attrition collection, utilize the aggregate function to group documents by Department(using Department field), match and count the number of employees in Sales department having a technical degree(using Educationfield field)

Screen Shot of MongoDB Query/Code and Results

Code:

```
db.attrition.aggregate([{"$match":{$and:[{Department:"Sales"},{EducationField:"Technical Degree"}]}},{"$group":{_id:"$Department",count:{$sum:1}}}])
```



Mongo Query 3

Question

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

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

- We assume that by comparing the average values of job satisfaction and environment satisfaction across different departments we can find out the department with highest job and environment satisfaction.
- We use the aggregate function to group the employees by departments and get the average values of job satisfaction and environment satisfaction.
- From the results, we can see that the Sales department has the highest average job satisfaction value of 2.751 compared to other two departments. It is also clear that the Research and development department has the highest average environment satisfaction value of 2.744.
- Therefore, **it is right that the Sales department has the highest job satisfaction while Research & Development department has the highest environment satisfaction**.

Translation

Using attrition collection, utilize aggregate function to group documents by Department (using Department field) and get the average of job satisfaction (using JobSatisfaction field) and average of environment satisfaction (using EnvironmentSatisfaction field) for the employees in each department.

Screen Shot of MongoDB Query/Code and Results

Code:

```
db.attrition.aggregate([{"$group":{_id:"$Department",AverageJobSatisfaction:{$avg:"$JobSatisfaction"},AverageEnvironmentSatisfaction:{$avg:"$EnvironmentSatisfaction"}}}])
```

```
> db.attrition.aggregate([{"$group":{_id:"$Department",AverageJobSatisfaction:{$avg:"$JobSatisfaction"},
AverageEnvironmentSatisfaction:{$avg:"$EnvironmentSatisfaction"}}}])
{ "_id" : "Sales", "AverageJobSatisfaction" : 2.7511210762331837, "AverageEnvironmentSatisfaction" : 2.6
79372197309417 }
{ "_id" : "Research & Development", "AverageJobSatisfaction" : 2.7263267429760667, "AverageEnvironmentSa
tisfaction" : 2.7440166493236213 }
{ "_id" : "Human Resources", "AverageJobSatisfaction" : 2.6031746031746033, "AverageEnvironmentSatisfact
ion" : 2.6825396825396823 }
>
```