Python and SQL project(Adventurework)

Data source

AdventureWorks database supports standard online transaction processing scenarios for a fictitious bicycle manufacturer - **Adventure Works Cycles**. Scenarios include Manufacturing, Sales, Purchasing, Product Management, Contact Management, and Human Resources.

Introduction

In this project we take advantages of the tools like Azure data studio, SQL Server Management Studio (SSMS), excel and python to observe the dataset and visualize our findings

- Azure Data Studio : Use query to retrieve the data from database and generate csv file
- SMSS : Draw the entity-relationship diagram, understand the primary key and foreign key of the database.
- Excel : Visualize the graph from the dataset we got in an Advance method
- Python : Use to import the csv file we have from azure data studio, and plot the graph to visualize the finding.

  ### Below is the library we used in Python

  - Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python.
  - NumPy is a Python library used for working with arrays. It offers comprehensive mathematical functions, random number generators, linear algebra routines, Fourier transforms, and matrices.
  - Pandas is a software library written for the Python programming language. for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series.
    .

▼ 1. Determining the regional sales in the best-performing country

Overview

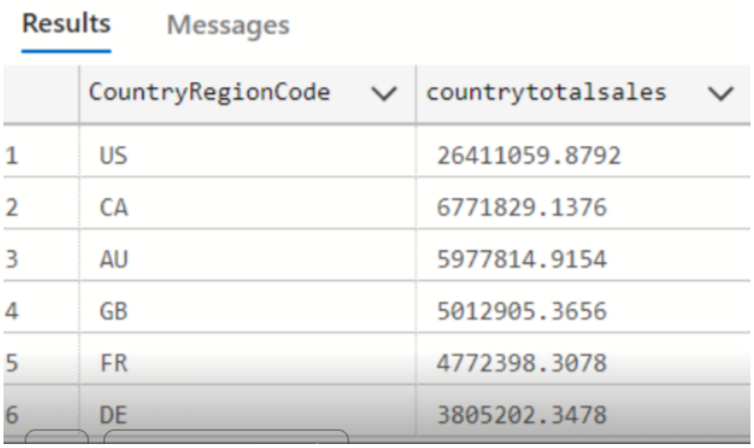
We used the SQL code to extract the tables showing sales for different regions. Then we used Python codes to compare the performances of these regions.

Retrieving the data

From the database, we observed that we will do the evaluation in two step:

First, we calculate the region sales and group by country, adding the aggregate function to sum the sales from each region where they belongs to the same country.

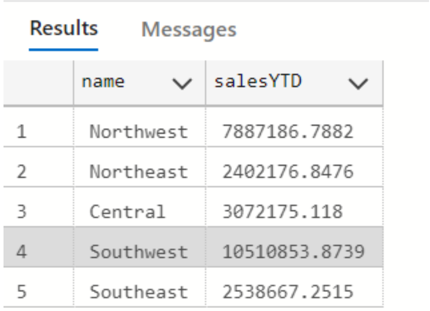
```
Select CountryRegionCode, sum(SalesYTD) countrytotalsales
from Sales.SalesTerritory
group by CountryRegionCode
order by countrytotalsales desc
```



	CountryRegionCode	countrytotalsales
1	US	26411059.8792
2	CA	6771829.1376
3	AU	5977814.9154
4	GB	5012905.3656
5	FR	4772398.3078
6	DE	3805202.3478

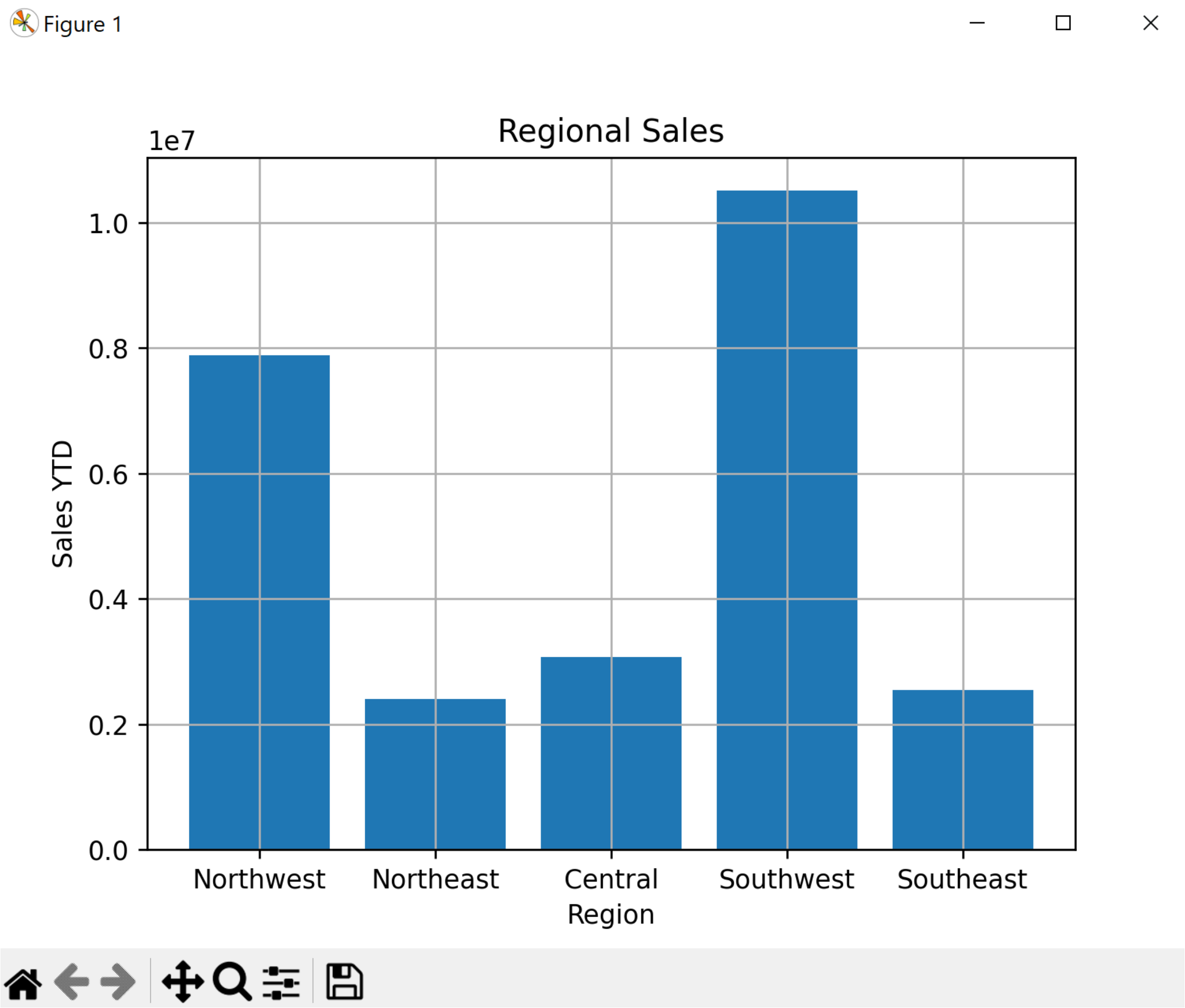
Secondly, from the previous query, we discovered that US has best-performing sales across all countries, then we proceeded the following code to find out the regional sales in the US to verify why this country perform so well.

```
Select name, salesYTD from sales.SalesTerritory
where CountryRegionCode = 'US'
```



	name	salesYTD
1	Northwest	7887186.7882
2	Northeast	2402176.8476
3	Central	3072175.118
4	Southwest	10510853.8739
5	Southeast	2538667.2515

```
import matplotlib.pyplot as plt
name = ['Northwest', 'Northeast', 'Central', 'Southwest', 'Southeast']
salesYTD = [7887186.7882, 2402176.8476, 3072175.118, 10510853.8739, 2538667.2515]
plt.bar(name, salesYTD)
plt.grid()
plt.xlabel('Region')
plt.ylabel('Sales YTD')
plt.title('Regional Sales')
plt.show()
```



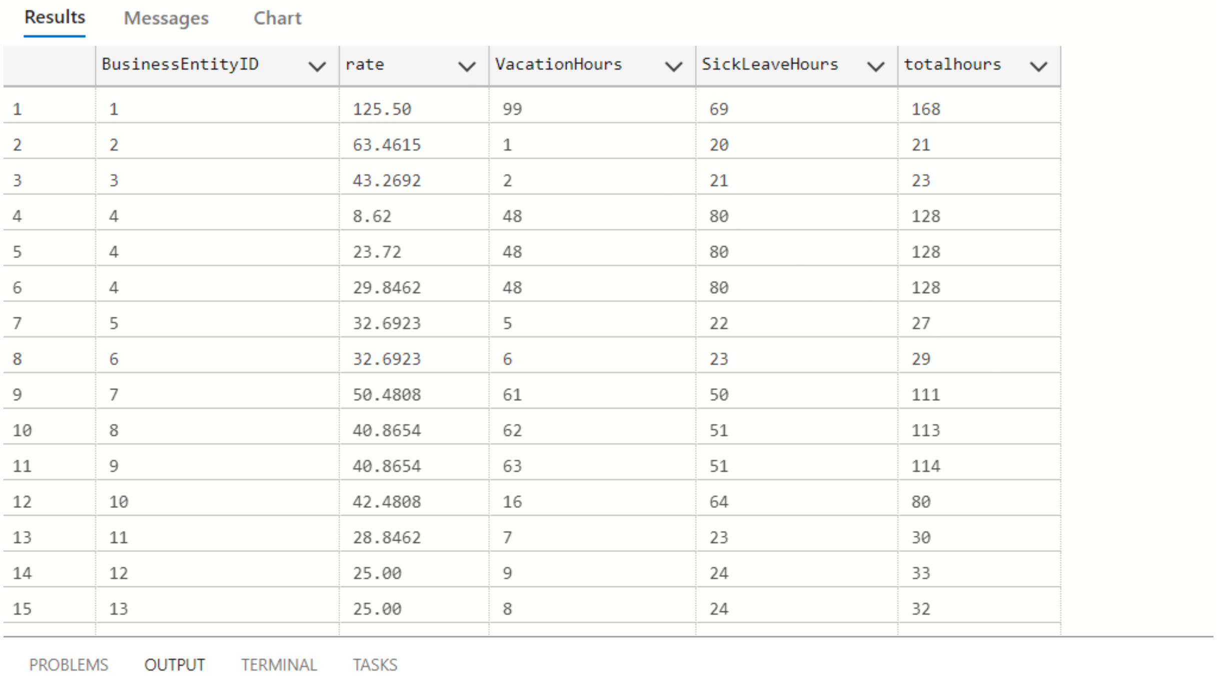
The bar chart shows the regional sales in US which is the best performing country. This chart shows that the Southwest region is the best performing region in the US, with the Southwest of the US outperforming complete countries, while the Southeast performing the weakest, this may be due to a myriad of reasons, therefore, using these results in conjunction with information about the reasons can be extremely useful in increasing sales not just in the US but internationally.

▼ 2. Finding the relationship between annual leave taken and bonus

Overview

We used SQL codes to extract a table showing the Business EntityID, Rate, Vacation Hours, Sick Leave Hours and Total Hours.

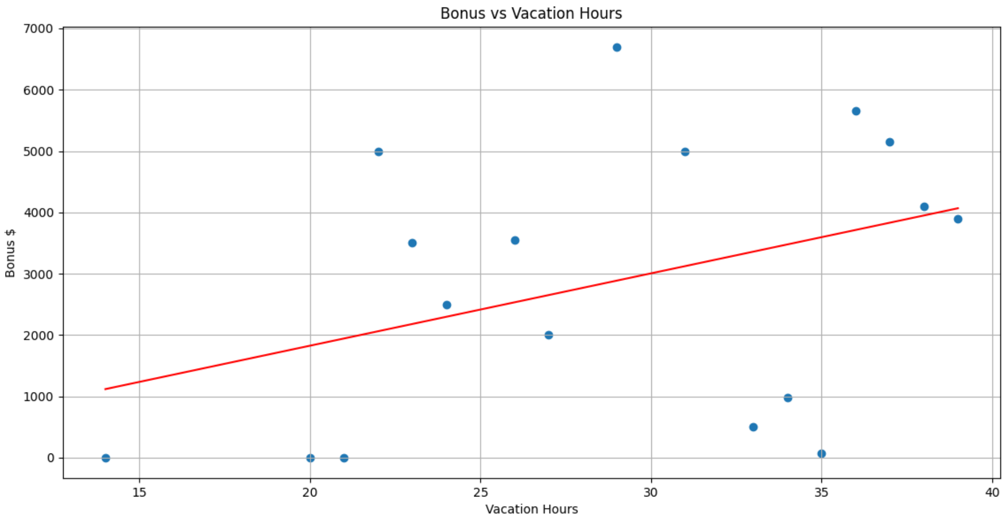
```
select a.BusinessEntityID,rate,VacationHours,SickLeaveHours,
(VacationHours+SickLeaveHours) as totalhours
from HumanResources.Employee a
inner JOIN HumanResources.EmployeePayHistory b
on a.BusinessEntityID=b.BusinessEntityID
```



	BusinessEntityID	rate	VacationHours	SickLeaveHours	totalhours
1	1	125.50	99	69	168
2	2	63.4615	1	20	21
3	3	43.2692	2	21	23
4	4	8.62	48	80	128
5	4	23.72	48	80	128
6	4	29.8462	48	80	128
7	5	32.6923	5	22	27
8	6	32.6923	6	23	29
9	7	50.4808	61	50	111
10	8	40.8654	62	51	113
11	9	40.8654	63	51	114
12	10	42.4808	16	64	80
13	11	28.8462	7	23	30
14	12	25.00	9	24	33
15	13	25.00	8	24	32

Then we plotted a scatter graph to visually verify the relationship between Bonus and Vacation hours using python code.

```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
from matplotlib.pyplot import figure
df = pd.read_csv(r"C:\Users\ahmed_pppk2gj\Downloads\q2.csv")
print (df)
plt.scatter(df.VacationHours, df.Bonus)
plt.xlabel('Vacation Hours')
plt.ylabel('Bonus $')
plt.title('Bonus vs Vacation Hours')
coefficients = np.polyfit(df.VacationHours, df.Bonus, 1)
polynomial = np.poly1d(coefficients)
x_points = np.linspace(min(df.VacationHours), max(df.VacationHours))
plt.plot(x_points, polynomial(x_points), '-r')
plt.grid()
plt.show()
```



Some of the points show there is some degree of positive relationship between Bonus and Vacation Hours, where more vacation hours indicated more bonuses earned. However some points do not conform to this as they completely contradict this relationship. Some points are close to the line of best fit.

Some points are far from the line of best fit , so they could be outliers as they do not follow the spread of the data however this may be due to the small sample size.

If the points are closer to the line of best fit, there would have been a stronger correlation which would imply a stronger positive relationship.

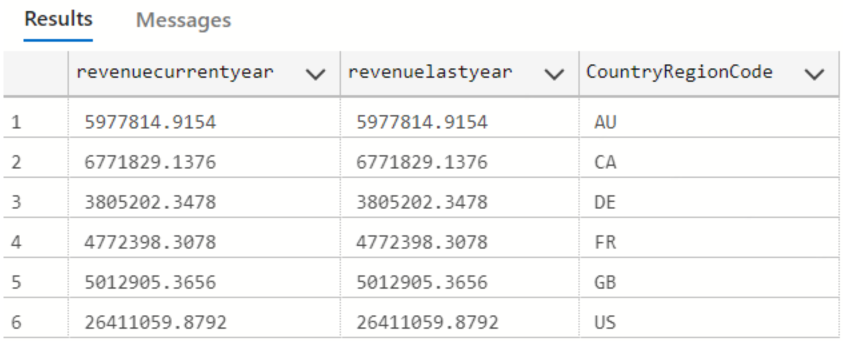
We concluded that there is a very weak positive relationship between Bonus and Vacation Hours because most of the points are not close to the line of best fit.

▼ 3. Determining the relationship between Country and Revenue

Overview

We used SQL codes to select the columns that shows revenue from the countries.

```
SELECT sum(SalesYTD) revenuecurrentyear, sum(SalesYTD) revenuelastyear, CountryRegionCode
  FROM [AdventureWorks2019].[Sales].[SalesTerritory]
  group by CountryRegionCode
```



	revenuecurrentyear	revenuelastyear	CountryRegionCode
1	5977814.9154	5977814.9154	AU
2	6771829.1376	6771829.1376	CA
3	3805202.3478	3805202.3478	DE
4	4772398.3078	4772398.3078	FR
5	5012905.3656	5012905.3656	GB
6	26411059.8792	26411059.8792	US

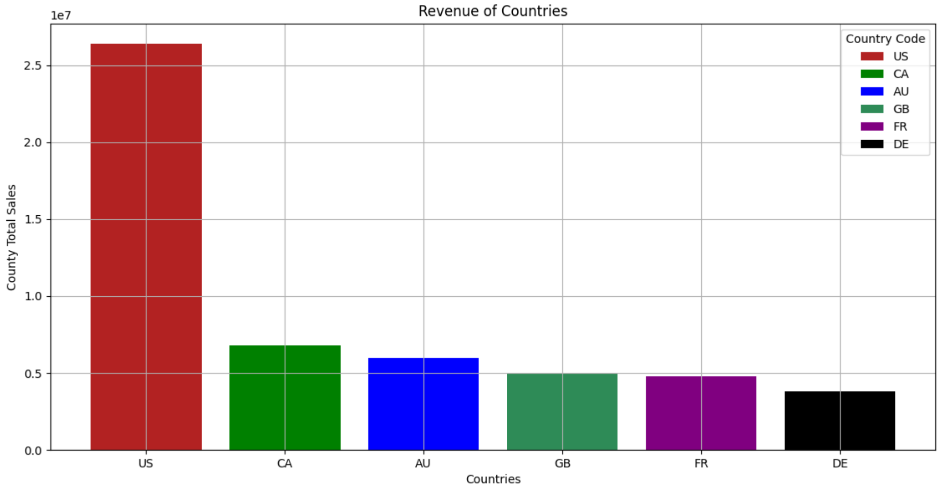
The table generated was converted to csv file and exported to python. Then a bar chart was plotted in python to show the relationship from various countries based on their sales revenue of the year to date.

▼ Visualization output

▼ Visualization for bar chart

```
#import libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.pyplot import figure

df = pd.read_csv(r'C:\projectgen\questionthree.csv')
print (df)
plt.bar(df.countryregioncode, df.countrytotalsales, color=['firebrick', 'green', 'blue', 'seagreen','purple', 'black'], label=
plt.xlabel('Countries')
plt.ylabel('County Total Sales')
plt.title('Revenue of Countries')
plt.grid()
plt.legend(title = 'Country Code')
plt.show()
```



▼ Visualization for pie chart

For additional visualisation, we plotted a pie chart to determine the relationship between various countries and their revenue.

From the two visualisations, we found out that the United States is the best-performing country, accounting for more than half of the whole sales followed by Canada while the European countries have the lowest revenue probably due to a lack of favourable policies to support growth.

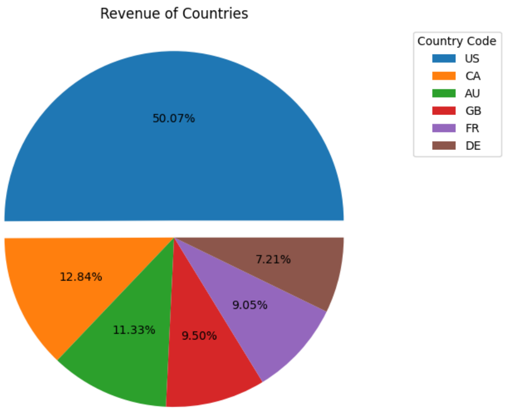
```
#import libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.pyplot import figure

df = pd.read_csv(r'C:\projectgen\questionone.csv')
print (df)

mylabels = [df.countryregioncode]
myexplode = [0.1, 0, 0, 0, 0, 0]
total = sum(df.countrytotalsales)

plt.pie(df.countrytotalsales,  explode = myexplode, autopct='%1.2f%%')
plt.legend(title = 'Country Code', labels = df.countryregioncode, bbox_to_anchor = (1.05, 1), loc= 'upper left')

plt.title('Revenue of Countries')
plt.show()
```



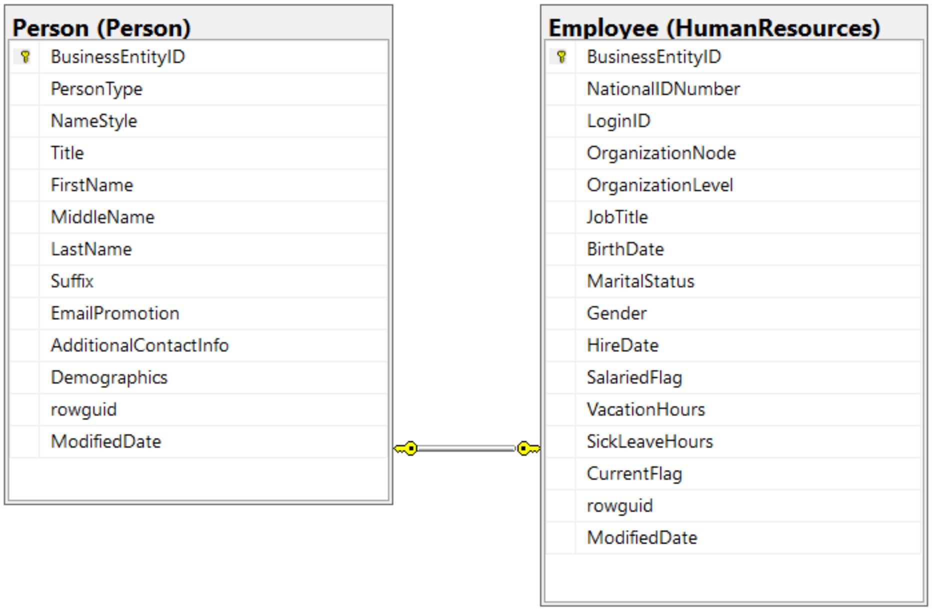
▼ 4. The relationship between sick leave and Job Title (Person Type)

Overview

To identify the relationship we first need to decide which table to use is the best fit for our finding.

Table relationship and try-and-error method is introduced to identify the right data to illustrate.

Afterwards, SQL code from only Human resources schema is used to query the data and convert to csv file for further visualization.



Person (Person)

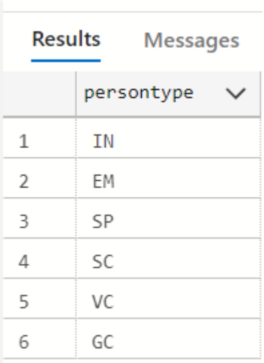
- BusinessEntityID
- PersonType
- NameStyle
- Title
- FirstName
- MiddleName
- LastName
- Suffix
- EmailPromotion
- AdditionalContactInfo
- Demographics
- rowguid
- ModifiedDate

Employee (HumanResources)

- BusinessEntityID
- NationalIDNumber
- LoginID
- OrganizationNode
- OrganizationLevel
- JobTitle
- BirthDate
- MaritalStatus
- Gender
- HireDate
- SalariedFlag
- VacationHours
- SickLeaveHours
- CurrentFlag
- rowguid
- ModifiedDate

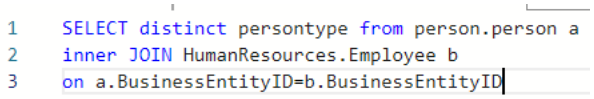
The following entity relationship we created have shown the Business entity ID are the primary keys for both table, and we will need Persontype from Left table while sick hour from right table.

At the first glance at the left table, we got 6 distinct person type. As shown as graph below.

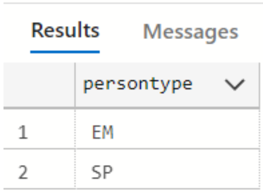


	persontype
1	IN
2	EM
3	SP
4	SC
5	VC
6	GC

After having a understanding on the left table, we inner join the right table to see how many person type having the data of sick leave hour.



```
SELECT distinct persontype from person.person a
inner JOIN HumanResources.Employee b
on a.BusinessEntityID=b.BusinessEntityID
```

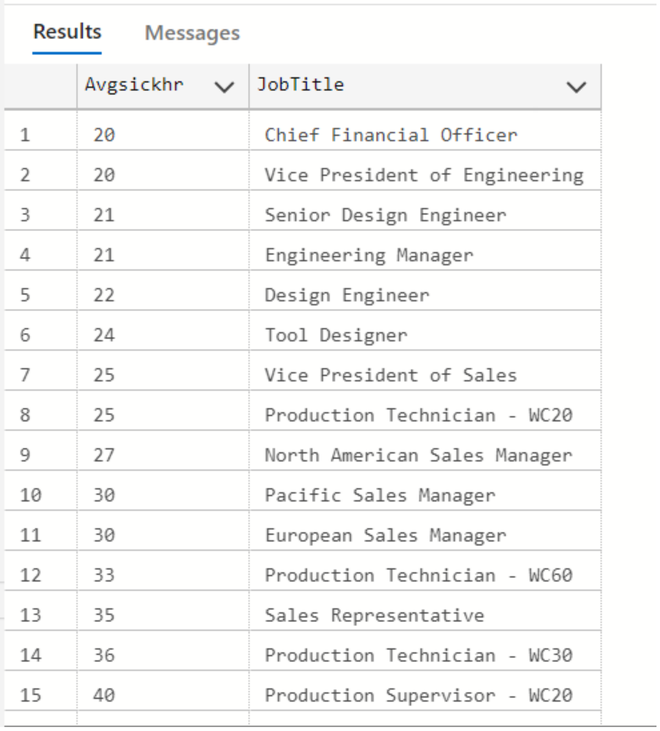


	persontype
1	EM
2	SP

The result shows that only EM and SP have data on sick leave hour, due to the incomprehensive person type after the joining. To avoid unfair data analysis due to the lack of full image, we decided to move on to the relationship between sick leave hour and job-title.

We queried the data and the result is shown below.

```
select avg(SickLeaveHours) Avgsickhr, JobTitle from HumanResources.Employee e
group by JobTitle
order by Avgsickhr
```



	Avgsickhr	JobTitle
1	20	Chief Financial Officer
2	20	Vice President of Engineering
3	21	Senior Design Engineer
4	21	Engineering Manager
5	22	Design Engineer
6	24	Tool Designer
7	25	Vice President of Sales
8	25	Production Technician - WC20
9	27	North American Sales Manager
10	30	Pacific Sales Manager
11	30	European Sales Manager
12	33	Production Technician - WC60
13	35	Sales Representative
14	36	Production Technician - WC30
15	40	Production Supervisor - WC20

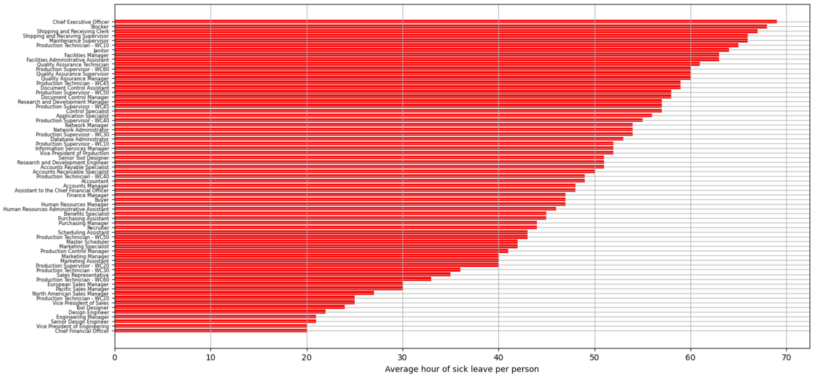
▼ Visualization code

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.pyplot import figure
```

```
plt.rc('ytick', labelsize= 6)
df = pd.read_csv(r'C:\Users\cylia\Desktop\BNO\DA bootcamp\project1\q4.csv')
x= list(df.iloc[:,1])
y= list(df.iloc[:,0])
plt.barh(x,y,color = 'r')
plt.grid()
plt.xlabel("Average hour of sick leave per person")
plt.ylabel("Job Title")
plt.show()
```

The code above use the libraries to plot a horizontal bar chart.

▼ Visualization output



The graph shows the relationship between job title and the average hour of sick leave hour per person per year.

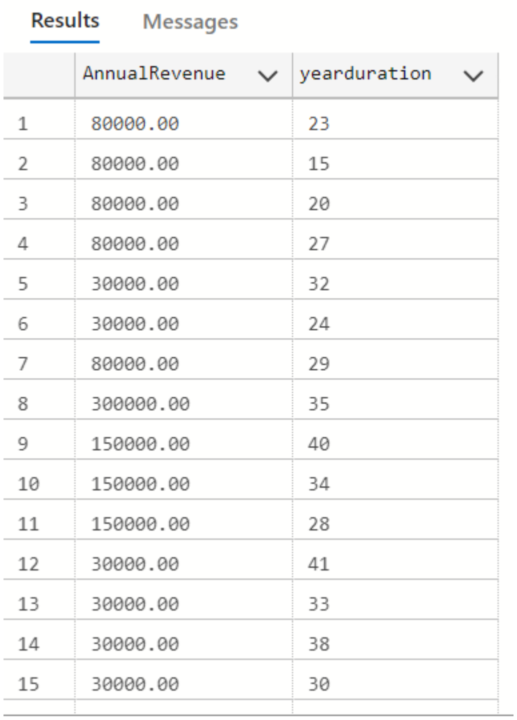
It is worth investigating that the CEO have the most sick leave hours per year on average while the CFO had the least, Further investigation behind the scenario should be conducted as while the CEO represent a important image of the company and is essentially the figurehead, they aren't as involved in the day to day running of the organisation and are at more liberty to take sick days. The visualisation also suggests a general trend that Supervisors involved within the sales tend to have less sick hours than those within logistics, while those in job roles main production such as quality assurance on average take more sick leave, however this may be down to a number of different factors, so while this information is useful, more will be needed in order to use this information to its full potential.

▼ 5. The relationship between store trading duration and revenue

Overview

```
select d.AnnualRevenue, DATEDIFF(y,YearOpened,2014) as yearduration from
[AdventureWorks2019].[Sales].[vStoreWithDemographics] d
inner join Sales.Store s
on d.BusinessEntityID=s.BusinessEntityID
```

We found a view that created from the database is efficient for our team to analyze the data, therefore, we join the view table(vstorewithdemographics) to analyze the data. The private key they share is business entity ID.



	AnnualRevenue	yearduration
1	80000.00	23
2	80000.00	15
3	80000.00	20
4	80000.00	27
5	30000.00	32
6	30000.00	24
7	80000.00	29
8	300000.00	35
9	150000.00	40
10	150000.00	34
11	150000.00	28
12	30000.00	41
13	30000.00	33
14	30000.00	38
15	30000.00	30

▼ Visualization code 1(showing the correlation coefficient value with scatterplot)

```
import matplotlib.pyplot as plt
import pandas as pd


data=pd.read_csv(r'C:\Users\cylia\Desktop\BNO\DA bootcamp\project1\q5.csv')
annual_revenue= list(data.iloc[:,0])
store_duration= list(data.iloc[:,1])

corr = data["yearduration"].corr(data["AnnualRevenue"])
# Create the scatter plot
plt.scatter(store_duration, annual_revenue)
# Add axis labels
plt.xlabel('Store Duration (years)')
plt.ylabel('Annual Revenue (USD)')
# Show the plot
plt.text(x=data["yearduration"].min(),y=data["AnnualRevenue"].max(),s="correlation
                 coefficient: {:.2f}".format(corr),fontsize=8)


plt.show()
```

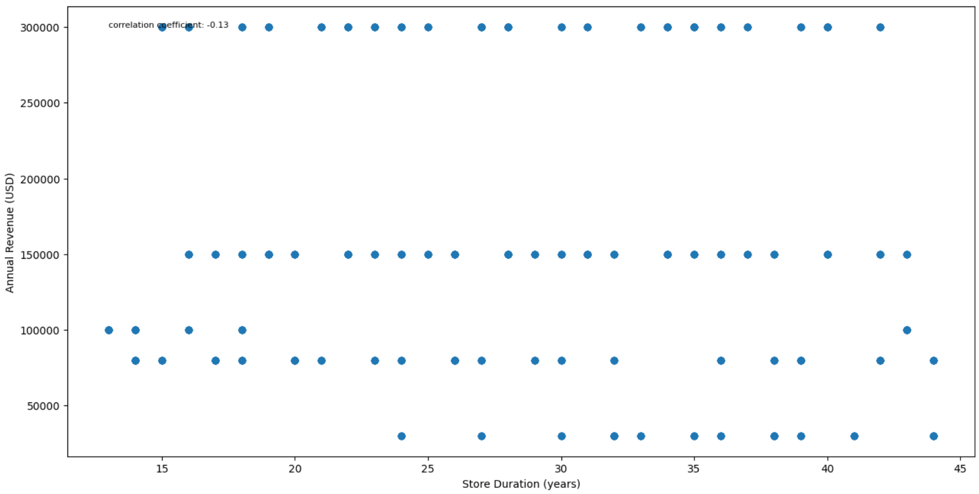
The above code used scatter plot and a text with correlation to visualize the data.

▼ Visualization code 2(showing the trendline with scatterplot)

```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
from matplotlib.pyplot import figure
df = pd.read_csv(r"C:\Users\ahmed_pppk2gj\Downloads\q5.csv")
print (df)
plt.scatter(df.AnnualRevenue, df.yearduration)
plt.xlabel('Revenue $')
plt.ylabel('Duration in Years')
plt.title('Relationship between Duration & Revenue')
coefficients = np.polyfit(df.AnnualRevenue, df.yearduration, 1)
```

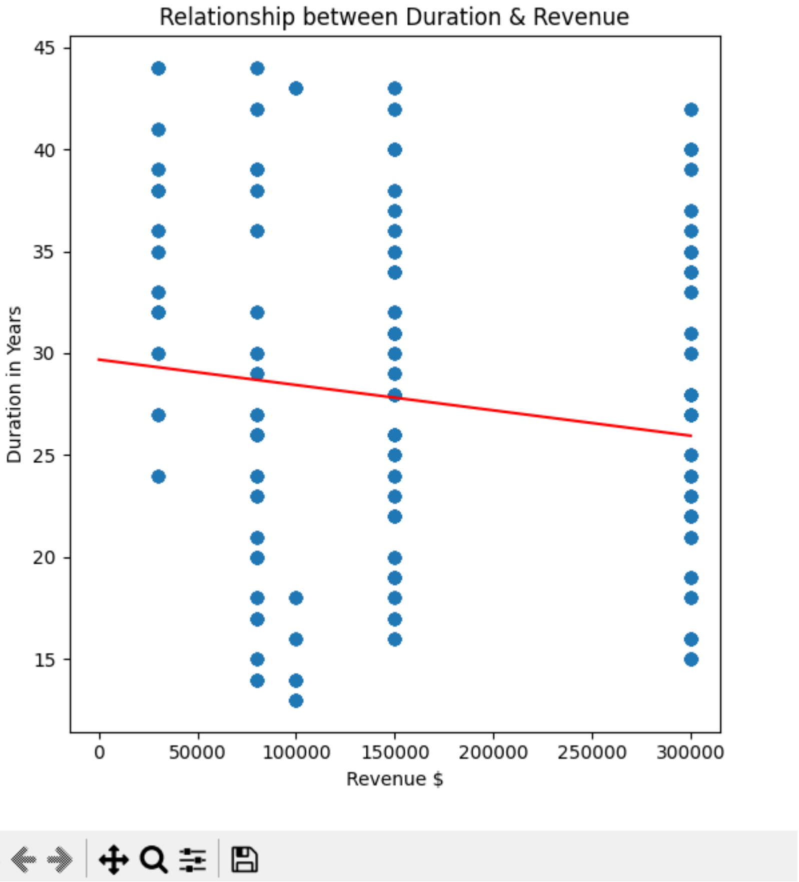
```
polynomial = np.poly1d(coefficients)
x_points = np.linspace(min(df.yearduration), max(df.AnnualRevenue))
plt.plot(x_points, polynomial(x_points), '-r' )
plt.show()
```

▼ Visualization output 1



The output 1 shows the correlation coefficient in text, it was shown the result is closed to 0, with a value with - 0.13, therefore, there is barely a noticeable relationship between duration and annual revenue.

▼ Visualization output 2



We used the data to find a relationship between store trading duration and revenue.
We decided to focus on annual revenue (USD) and Duration on years the Stores business has been open.
Our results showed that there was large range of data. We decided that the most appropriate graph would be a scatter plot with a line of best fit.

Using matplotlib and pandas through the python interface we imported the data via CSV file. Our hypotheses were correct, and the data was shown to be a very weakly negative correction.

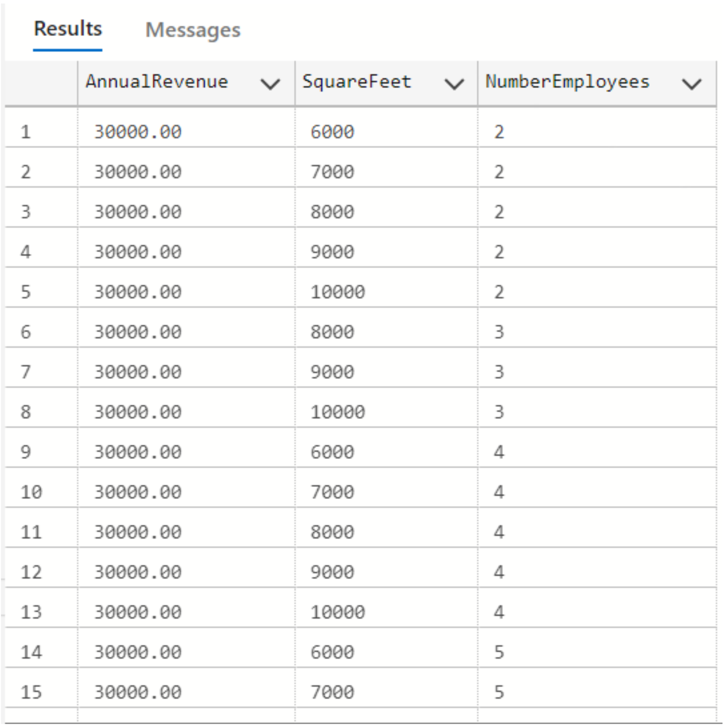
Whether or not how old the business is. The annual revenue can range from $60,000 - $300,000. This is at 15 years, the spread of results make it difficult to make a conclusion based on these relationships of data

We used a range of software including VSC and EXCEL to confirm our results

▼ 6. The relationship between the size of the stores, number of employees and revenue.

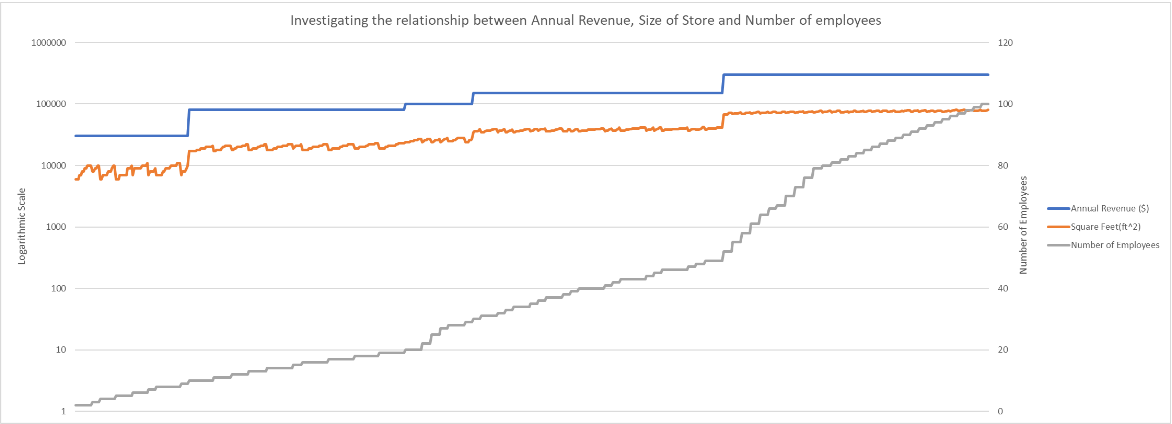
Overview

```
select distinct d.AnnualRevenue, SquareFeet, NumberEmployees
from [AdventureWorks2019].[Sales].[vStoreWithDemographics] d
order by AnnualRevenue, NumberEmployees,SquareFeet
```



	AnnualRevenue	SquareFeet	NumberEmployees
1	30000.00	6000	2
2	30000.00	7000	2
3	30000.00	8000	2
4	30000.00	9000	2
5	30000.00	10000	2
6	30000.00	8000	3
7	30000.00	9000	3
8	30000.00	10000	3
9	30000.00	6000	4
10	30000.00	7000	4
11	30000.00	8000	4
12	30000.00	9000	4
13	30000.00	10000	4
14	30000.00	6000	5
15	30000.00	7000	5

▼ Visualization output

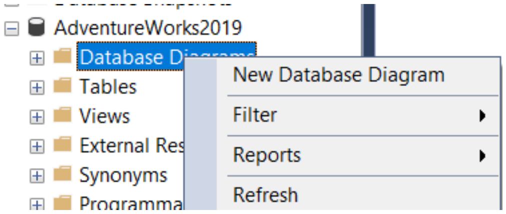


As you can see here, the two lines that represent the total revenue and the square foot of each store are almost parallel and even mirror each other which suggests that the square footage of the store has a direct link to the amount of revenue produced, an important thing to note is that this graph has a Logarithmic scale and is going up in an exponential manner. The number of employees plays an effect on the total revenue as a general relationship is a strong positive trend as the number of employees increases so does the revenue, and in turn the square footage also increases.

Interestingly it can be seen that this trend began to slow and began to plateau when the number of employees overlapped on the square footage line, as this suggests that it began to no longer have as much of an effect on the sales revenue.

Entity-relationship analysis

To understand the database deeply, it is essential to analyse the relationship we used SMSS server to generate the whole diagram.



This can be simply down by clicking the below button in image.

The actual diagram generated will be documented in appendix as Figure 1.

The AdventureWorks2019 database is a sample database provided by Microsoft that represents a fictitious company called Adventure Works Cycles. The database has several entities, including:

- Person: Contains information about individuals associated with the company, such as employees, customers, and vendors.
- Sales: Contains information about sales orders, sales order details, and sales territories.
- Production: Contains information about products, production plans, work orders, and bill of materials.
- Human Resources: Contains information about employee information, job candidates, and employee payroll.
- Purchasing: Contains information about purchase orders, vendor information, and product receipts.

These entities have relationships with each other, such as:

- A person can be an employee, customer, or vendor.
- An employee can have multiple job candidates.
- A sales order is related to a sales territory and to a customer.
- A product is related to a production plan and to a bill of materials.
- A purchase order is related to a vendor and to a product receipt.

These relationships are used to model the relationships between different entities in the AdventureWorks2019 database and to ensure data integrity.

Conclusion

This project while challenging especially due to time constraints and unforeseen circumstances was a fantastic opportunity for our team to improve our skills and use a wide range of skills that would be utilised by a data analyst in there day to day. We feel as a team we have grabbed this opportunity by our two hands, and have illustrated a snippet of what we can do, by answering the questions put before us accurately and efficiently using skills which are integral to being a data analyst. In order for our findings to be even more useful to the company, a wide range of further investigation and research must be needed such as why the Southwest has the largest sales for the company, why the sales in European countries are lower than that of North America, and the correlation between upper management and more increased sick leave per year on average. But this investigation provides an excellent baseline to build off of.

Appendix

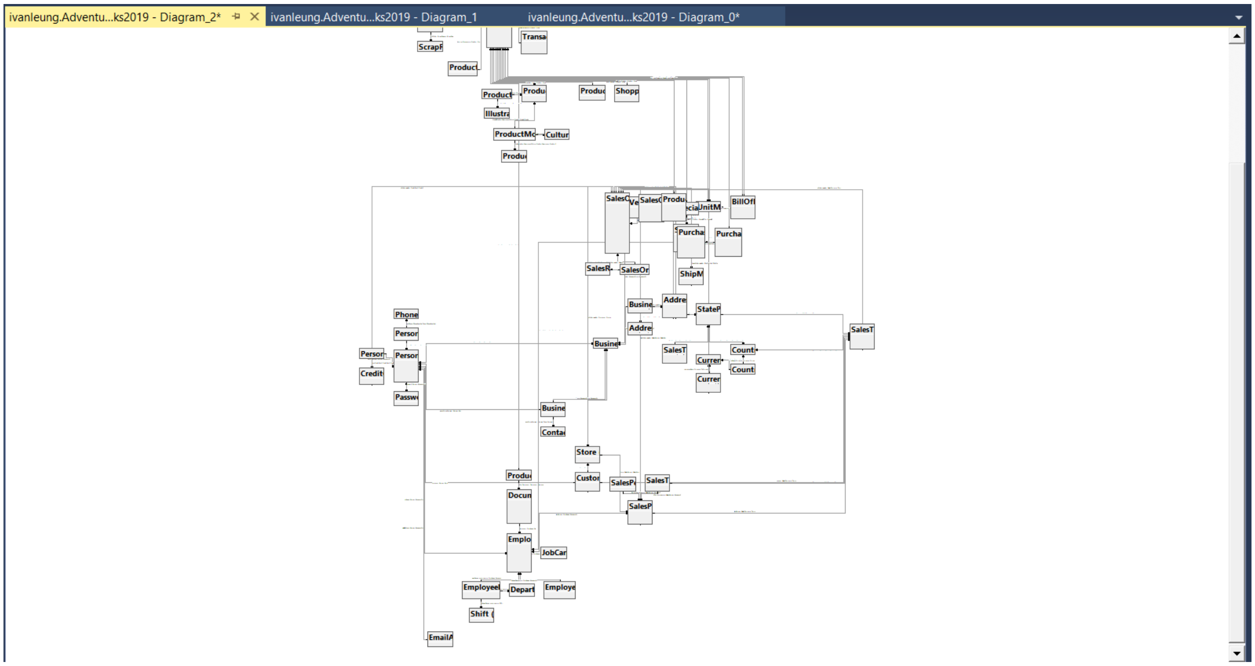


Figure 1- Database schema