**Team 2**

**Data Analytics Interim Project Proposal**

**TEMPLATE**

**Overview**

*Use SSMS to create views from the adventureworks database with the required data which will then be pulled into python to be analysed and visually displayed.*

|  |  |
| --- | --- |
| Proposed by: | Team Number: 2 |
| Timeframe: | Completion by 31/05/2023 23:59  Presentation on 01/06/2023 |

**Objectives and steps**

With the use of specific data found in Adventure works we were tasked to answer the following questions: *[for each question below, add descriptions of your groups’ ideas generated, conceptual sketches, function, and key components you will use]*

Python Libraries used

· pyodbc https://pypi.org/project/pyodbc/

· pandas (imported as pd) https://pandas.pydata.org

· matplotlib (specifically matplotlib.pyplot imported as plt) https://matplotlib.org

1. What is the regional sales in the best performing country?

Creating a view that pulls in the country region code (CountryRegionCode) as Country from the Sales.SalesTerritory table the sum of subtotal from the Sales.SalesOrderHeader as Sales this will require a join on Sales.SalesTerritory (SalesOrderHeader.TerritoryID = SalesTerritory.TerritoryID). Furthermore this will be grouped by CountryRegionCode and ordered by Sales in descending order.

As US was the best performing country a second view was created to analyse its regions this view will be the exact same as the above but only we will filter the query to only return values with the CountryRegionCode of ‘US’. This will then be grouped by Name and Sales in descending order.

This view can then be brought into python by the use of the Pyodbc package in which it can then be analysed with packages such as matplotlib, pandas and numpy. We opted to make two bar charts to visualise the answers of this question.

1. What is the relationship between annual leave taken and bonus?

Creating a view that pulls in business entity ID and vacation hours from HumanResources.Employee table, the bonus from the Sales.SalesPerson table this will require a join of the Sales.SalesPerson table on the BusinessEntityID, this view can then be brought into python by the use of the Pyodbc package in which it can then be analysed with packages such as matplotlib, pandas and numpy. Specifically we will be looking for the correlation to accomplish this we opted to use a scatter graph to visualise this.

1. What is the relationship between Country and Revenue?

Creating a view that pulls in Country Name (Name), the sum of SalesLastYear and SalesYTD (call this total sales) from Sales.SalesTerritory. We will group this view by the country name. This view can then be brought into python by the use of the Pyodbc package in which it can then be analysed with packages such as matplotlib, pandas and numpy. We opted to visualise this data in a bar graph and pie chart.

1. What is the relationship between sick leave and Job Title?

Creating a view that pulls in an employees job title (JobTitle) and the sick leave hours (SickLeaveHours) and OrganizationLevel from HumanResources.Employee. We will group this data by OrganizationalLevel and JobTitle as well as order it by OrganizationalLevel and AverageSickLeave in descending order. This view can then be brought into python by the use of the Pyodbc package in which it can then be analysed with packages such as matplotlib, pandas and numpy. We opted to visualise this data in 2 bar graphs.

1. What is the relationship between store trading duration and revenue?

Creating a view that pulls in store name (StoreName), Year Opened (YearOpened), AnnualRevenue from Sales.vStoreWithDemographics for the trading duration (2023 - YearOpened). A join will then need to be done on Sales.Customer (vStoreWithDemographics.BusinessEntityID = Customer.StoreID) so that stores and the sales included in Sales.SalesOrderHeader can be associated (this will be our second join Customer.CustomerID = SalesOrderHeader.CustomerID) by doing this the view has access to the overall revenue that we get by summing TotalDue from the SalesOrderHeader table. We will then group this view by StoreName, YearOpened, TradingDuration and YearlyRevenue This view can then be brought into python by the use of the Pyodbc package in which it can then be analysed with packages such as matplotlib, pandas and numpy. We opted to visualise this data with two scatter plots.

1. What is the relationship between the size of the stores, number of employees and revenue?

Creating a view that pulls in store name (StoreName), Store Size (SquaredFeet), Number of Employees (NumberEmployees), and AnnualRevenue from Sales.vStoreWithDemographics. Join will then need to be done on Sales.Customer (vStoreWithDemographics.BusinessEntityID = Customer.StoreID) so that stores and the sales included in Sales.SalesOrderHeader can be associated (this will be our second join Customer.CustomerID = SalesOrderHeader.CustomerID) by doing this the view has access to the overall revenue that we get by summing TotalDue from the SalesOrderHeader table. We will then group this view by StoreName, StoreSize, TotalEmployees and YearlyRevenue. This view can then be brought into python by the use of the Pyodbc package in which it can then be analysed with packages such as matplotlib, pandas and numpy. We opted to visualise this data with multiple scatter plots.