```
/*


                    AdventureWorks2019 Employee Analysis

        Original tables that will be used for the analysis

SELECT * FROM HumanResources.Department;

SELECT * FROM HumanResources.Employee;

SELECT * FROM HumanResources.EmployeeDepartmentHistory;

SELECT * FROM HumanResources.EmployeePayHistory;

SELECT * FROM Person.Person;

Note: Not all the queries written here were used in Power BI
*/


-- Clean Person table by replacing NULL values and selecting relevant columns and rows for analysis
SELECT
    BusinessEntityID,
    PersonType,
    ISNULL(Title, 'MISSING') AS Title,
    [FirstName],
    ISNULL(MiddleName, 'N/A') AS MiddleName,
    LastName
INTO [Person].[Person - Cleaned]    -- Puts query into a new table
FROM [Person].[Person]
WHERE PersonType IN ('EM', 'SP');   -- Added to BI

-- Test to see if the new table displays output as intended
SELECT * FROM [Person].[Person - Cleaned];
```

```sql
-- Select relevant columns for analysis and put into a new table
SELECT
    BusinessEntityID,
    JobTitle,
    BirthDate,
    Gender,
    HireDate,
    VacationHours,
    SickLeaveHours,
    ModifiedDate
INTO [HumanResources].[Employee - Cleaned]
FROM [HumanResources].[Employee];   -- Added to BI

SELECT * FROM [HumanResources].[Employee - Cleaned]; -- Check to see if above query works properly


-- Count of Employees hired each year by gender and results sent to a virtual table
CREATE VIEW [Employees Hired]
AS
SELECT
    gender,
    SUM(CASE WHEN HireDate BETWEEN '1/1/2006' AND '12/31/2006' THEN 1  ELSE ' ' END) AS [2006 Total],
    SUM(CASE WHEN HireDate BETWEEN '1/1/2007' AND '12/31/2007' THEN 1 END) AS [2007 Total],
    SUM(CASE WHEN HireDate BETWEEN '1/1/2008' AND '12/31/2008' THEN 1 END) AS [2008 Total],
    SUM(CASE WHEN HireDate BETWEEN '1/1/2009' AND '12/31/2009' THEN 1 END) AS [2009 Total],
    SUM(CASE WHEN HireDate BETWEEN '1/1/2010' AND '12/31/2010' THEN 1 END) AS [2010 Total],
    SUM(CASE WHEN HireDate BETWEEN '1/1/2011' AND '12/31/2011' THEN 1 END) AS [2011 Total],
    SUM(CASE WHEN HireDate BETWEEN '1/1/2012' AND '12/31/2012' THEN 1 END) AS [2012 Total],
    SUM(CASE WHEN HireDate BETWEEN '1/1/2013' AND '12/31/2013' THEN 1 END) AS [2013 Total],
    COUNT(*) AS [Total Employees Hired]
FROM [HumanResources].[Employee - Cleaned]
group by gender;

SELECT * FROM [Employees Hired]; -- Check if virtual table displays intended output


-- Count of Employees hired by year
```

```sql
SELECT
    COUNT(CASE WHEN HireDate BETWEEN '1/1/2006' AND '12/31/2006' THEN 1 END) AS [2006 Total],
    COUNT(CASE WHEN HireDate BETWEEN '1/1/2007' AND '12/31/2007' THEN 1 END) AS [2007 Total],
    COUNT(CASE WHEN HireDate BETWEEN '1/1/2008' AND '12/31/2008' THEN 1 END) AS [2008 Total],
    COUNT(CASE WHEN HireDate BETWEEN '1/1/2009' AND '12/31/2009' THEN 1 END) AS [2009 Total],
    COUNT(CASE WHEN HireDate BETWEEN '1/1/2010' AND '12/31/2010' THEN 1 END) AS [2010 Total],
    COUNT(CASE WHEN HireDate BETWEEN '1/1/2011' AND '12/31/2011' THEN 1 END) AS [2011 Total],
    COUNT(CASE WHEN HireDate BETWEEN '1/1/2012' AND '12/31/2012' THEN 1 END) AS [2012 Total],
    COUNT(CASE WHEN HireDate BETWEEN '1/1/2013' AND '12/31/2013' THEN 1 END) AS [2013 Total],
    COUNT(*) AS [Total Employees Hired]
FROM [HumanResources].[Employee - Cleaned] -- Added to BI


-- Virtual table to contain a cleaned table of EmployeeDepartmentHistory
CREATE VIEW [EmployeeDepartmentHistory - Cleaned]
AS
SELECT
    BusinessEntityID,
    DepartmentID,
    StartDate,
    (CASE WHEN [EndDate] IS NULL THEN 'Employed'  ELSE 'INACTIVE' END) AS [EndDate],
    ModifiedDate
FROM [HumanResources].[EmployeeDepartmentHistory]; -- Added to BI

SELECT * FROM [EmployeeDepartmentHistory - Cleaned]; -- Test to see if above query works properly


-- Find people that have more than one row of pay
SELECT
    BusinessEntityID,
    MAX(Rate) AS Rate,
    COUNT(*) AS [Row Count]
FROM HumanResources.EmployeePayHistory
GROUP BY BusinessEntityID
HAVING COUNT(*) > 1; -- 13 employees have their name come up 3 times, could be due to promotion
```

```sql
-- Joined four tables to get relevant tables and columns for analysis and stored in a virutal table
CREATE VIEW [Employee Pay]
AS
SELECT
    EDH.BusinessEntityID,
    D.DepartmentID,
    D.Name,
    D.GroupName,
    PC.FirstName,
    PC.LastName,
    EC.gender,
    MAX(ROUND(EPH.Rate, 2)) AS Rate -- To get single max pay for each employee, gets rid of previous pay
FROM HumanResources.EmployeeDepartmentHistory EDH
INNER JOIN HumanResources.Department D
    ON EDH.DepartmentID = D.DepartmentID
INNER JOIN [Person].[Person - Cleaned] PC
    ON PC.BusinessEntityID = EDH.BusinessEntityID
INNER JOIN [HumanResources].[Employee - Cleaned] EC
    ON EC.BusinessEntityID = PC.BusinessEntityID
INNER JOIN HumanResources.EmployeePayHistory EPH
    ON EPH.BusinessEntityID = EC.BusinessEntityID
WHERE EndDate IS NULL
GROUP BY
    EDH.BusinessEntityID,
    D.DepartmentID,
    D.Name,
    D.GroupName,
    PC.FirstName,
    PC.LastName,
    EC.gender;  -- Added to BI

 SELECT * FROM [Employee Pay] -- Test to make sure the query above works as intended


 -- Moving pay average by department name
SELECT
    DepartmentID,
```

```
    Name AS [Department Name],
    AVG(Rate)
    OVER (PARTITION BY [Name]
                ORDER BY Rate)
        AS [Moving Pay Average by Department]
FROM [Employee Pay]
GROUP BY
    [Name],
    DepartmentID,
    Rate;


-- Count of wage by category
SELECT
        COUNT(CASE WHEN Rate BETWEEN 0 AND 15 THEN 1 END) AS [Low Total],
        COUNT(CASE WHEN Rate BETWEEN 15.01 AND 30 THEN 1 END) AS [Medium Total],
        COUNT(CASE WHEN Rate > 30 THEN 1 END) AS [High Total],
        COUNT(*) AS Total
FROM [Employee Pay]; -- Added to BI


-- Pay Category by gender
SELECT
    gender,
    SUM(CASE WHEN Rate BETWEEN 0 AND 15 THEN 1 END) AS [Low Total],
    SUM(CASE WHEN Rate BETWEEN 15.01 AND 30 THEN 1 END) AS [Medium Total],
    SUM(CASE WHEN Rate > 30 THEN 1 END) AS [High Total],
    COUNT(*) AS Total
FROM [Employee Pay]
GROUP BY gender;


-- Average pay by gender
SELECT
    gender,
    ROUND(AVG(Rate), 2) AS [Pay Average by Department]
FROM [Employee Pay]
```

```sql
GROUP BY gender;    -- Added to BI


-- Employee count by gender
SELECT
    gender,
    COUNT(*) AS Employees
FROM [Employee Pay]
GROUP BY gender;    -- Added to BI


-- Average pay by department name
SELECT
    DepartmentID,
    Name AS [Department Name],
    ROUND(AVG(Rate), 2) AS [Pay Average by Department]
FROM [Employee Pay]
GROUP BY DepartmentID, Name
ORDER BY 3 DESC;    -- Added to BI


-- Get highest pay, lowest pay, and average pay by gender
SELECT
    gender,
    MAX(Rate) AS [Highest Pay],
    MIN(Rate) AS [Lowest Pay],
    ROUND(AVG(Rate), 2) AS [Average Pay]
FROM [Employee Pay]
GROUP BY gender;


-- Top 10 people with highest pay
SELECT TOP 10
    FirstName,
    LastName,
    gender,
    MAX(Rate) AS [Highest Pay]
```

```sql
FROM [Employee Pay]
GROUP BY FirstName, LastName, gender
ORDER BY [Highest Pay] DESC;


-- Lowest 10 payed people
SELECT TOP 10
    FirstName,
    LastName,
    gender,
    MIN(Rate) AS [Lowest Pay]
FROM [Employee Pay]
GROUP BY
    FirstName,
    LastName,
    gender
ORDER BY [Lowest Pay] ASC;


-- Top 10 people with most vacation hours
SELECT TOP 10
    EP.FirstName,
    EP.LastName,
    EP.gender,
    EC.JobTitle,
    EP.Rate,
    MAX(EC.VacationHours) AS [Most Vaction Hours]
FROM [HumanResources].[Employee - Cleaned] EC
INNER JOIN [Employee Pay] EP
    ON EC.BusinessEntityID = EP.BusinessEntityID
GROUP BY
    EP.FirstName,
    EP.LastName,
    EP.gender,
    EC.JobTitle,
    EP.Rate
ORDER BY [Most Vaction Hours] DESC;
```

```sql
-- Top 10 people with most sick leave hours
SELECT TOP 10
    EP.FirstName,
    EP.LastName,
    EP.gender,
    EC.JobTitle,
    EP.Rate,
    MAX(EC.SickLeaveHours) AS [Most Sick Leave Hours]
FROM [HumanResources].[Employee - Cleaned] EC
INNER JOIN [Employee Pay] EP
    ON EC.BusinessEntityID = EP.BusinessEntityID
GROUP BY
    EP.FirstName,
    EP.LastName,
    EP.gender,
    EC.JobTitle,
    EP.Rate
ORDER BY [Most Sick Leave Hours] DESC;


-- Total vacation hours and sick leave hours by gender
SELECT
    gender,
    SUM(VacationHours) AS [Total Vacation Hours],
    SUM(SickLeaveHours) AS [Total Sick Leave Hours]
FROM [HumanResources].[Employee - Cleaned]
GROUP BY gender;    -- Added to BI



--People born since 1988 that are employed ordered by highest pay
SELECT
    EP.FirstName,
    EP.LastName,
    EP.gender,
```

```
    EC.BirthDate,
    EC.JobTitle,
    EP.Rate
FROM [HumanResources].[Employee - Cleaned] EC
INNER JOIN [Employee Pay] EP
ON EC.BusinessEntityID = EP.BusinessEntityID
WHERE EC.BirthDate >= '1/1/1988'
ORDER BY Rate DESC;
```