**Module 7 Challenge Report**

**(Francis Odo)**

**Background**

The management of Pewlett Packard wants to future-proof the company. In order to achieve this task, there are a number of steps that will need to be taken. However, knowing and understanding your employee skill-set and status is key to successful future decision making, and it starts with clean employee dataset from an updated database.

**Objective**

The objective is to analyze the current employee data, and:

1. Determine the number of employees per title who will be retiring and who’s eligible for retirement package.
2. Identify those employees who are eligible to participate in a mentorship program.

**Method and Approach**

The dataset provided for this analysis are a set of Comma Separated Value (CSV) files in tabular format. These files include employees.csv, departments.csv, dept\_emp.csv, salary.csv, title.csv and manager\_info.csv.

Our method of approach is to filter and extract as much information as necessary from the set of data provided and create more meaningful and easy to interpret sets of data, which will assist the management team in their decision making process.

**Code Plan**

1. Using the above referenced files provided, open and examine each table in the files and observe the relations between them.
2. Create an Entity Relations Diagram (ERD) that highlights what the tables have in common. This will be used as a guide for where to find what as we extract data from the files. See “EmployeeDB.PNG”.
3. Part 1. Create a table that gathers information from three tables (Employees, Salaries and Tittle). To achieve this, we need to perform INNER JOIN between the three tables mentioned. The resulting table will produce emp\_no, first\_name, last\_name, salary, title, from\_date and to\_date. The information obtained from this table will be exported as “retiring\_emp\_info.csv” Further, we group by title in a descending order. The grouped order by job title will produce “new\_title\_group.csv”.
4. Part 2. Two key requirements in this project is to group by title and retirement eligibility using the birth date criteria (Jan. 1, 1952 and Dec. 31, 1955). In addition to filtering, we also partition for better and easy assessment. The output from this table will be converted and exported as “unique\_title.csv” file.
5. Part 3. Create a new table that shows emp\_no, first\_name, last\_name, from\_date, to\_date and birth\_date. Perform INNER JOIN between employees and title tables from where to extract. Filter as necessary.

**Difficulty And Challenges**

Although, we have lots of information on the employees based on the data provided. The information exists in so many files with different tables. Tables with large data.

The data provided is not as clean. In other words, there are duplicates and missing data in some cases. Most evident is the fact that some employees held multiple titles over the years and the criteria for mentorship program eligibility not as clearly defined.

**Steps Taken To Overcome Challenges**

1. Extract relevant information into new tables by JOINING some existing tables.
2. Apply specific database techniques for filtering (WHERE, AND, ORDER BY)
3. Group by most recent title held by the employee using PARTITION, as well as, GROUP BY and COUNT methods. This way information is better organized and faster to retrieve.

**Conclusion / Observation**

Total Number of employees eligible for retirement =====72459

Grouped by job title:

|  |  |
| --- | --- |
| Senior Engineer | 25916 |
| Senior Staff | 24926 |
| Engineer | 9285 |
| Staff | 7636 |
| Technique Leader | 3603 |
| Assistant Engineer | 1090 |
| Manager | 2 |

Total number of employees that will be eligible for mentorship program ===== 1550

One can observe that the company has a lot of employees that will have to retire.

A screenshot of a cell phone

Description automatically generated

Entity Relations Diagram (ERD)

-- Creating tables for PH-EmployeeDB

CREATE TABLE departments (

     dept\_no VARCHAR(4) NOT NULL,

     dept\_name VARCHAR(40) NOT NULL,

     PRIMARY KEY (dept\_no),

     UNIQUE (dept\_name)

);

CREATE TABLE employees (

   emp\_no INT NOT NULL,

     birth\_date DATE NOT NULL,

     first\_name VARCHAR NOT NULL,

     last\_name VARCHAR NOT NULL,

     gender VARCHAR NOT NULL,

     hire\_date DATE NOT NULL,

     PRIMARY KEY (emp\_no)

);

CREATE TABLE dept\_manager (

dept\_no VARCHAR(4) NOT NULL,

  emp\_no INT NOT NULL,

  from\_date DATE NOT NULL,

  to\_date DATE NOT NULL,

FOREIGN KEY (emp\_no) REFERENCES employees (emp\_no),

FOREIGN KEY (dept\_no) REFERENCES departments (dept\_no),

  PRIMARY KEY (emp\_no, dept\_no)

);

CREATE TABLE salaries (

  emp\_no INT NOT NULL,

  salary INT NOT NULL,

  from\_date DATE NOT NULL,

  to\_date DATE NOT NULL,

  PRIMARY KEY (emp\_no)

);

CREATE TABLE dept\_emp (

  emp\_no INT NOT NULL,

  dept\_no VARCHAR (10) NOT NULL,

  from\_date DATE NOT NULL,

  to\_date DATE NOT NULL,

  PRIMARY KEY (emp\_no)

);

CREATE TABLE title (

  emp\_no INT NOT NULL,

  salary INT NOT NULL,

  from\_date DATE NOT NULL,

  to\_date DATE NOT NULL,

  PRIMARY KEY (emp\_no)

);