#### ***SQL Homework Write-Up***

Database Analytics Boot Camp

June 20, 2019

Prepared By:

James Dodson

Table of Contents

[I. Background 2](#_Toc12109331)

[II. Data Modeling 2](#_Toc12109332)

[III. Data Analysis 2](#_Toc12109333)

[1. List the following details of each employee: employee number, last name, first name, gender, and salary. 2](#_Toc12109334)

[1.1. SQL Statement: 2](#_Toc12109335)

[1.2. Query Result Set (Limit 5): 3](#_Toc12109336)

[2. List employees who were hired in 1986. 3](#_Toc12109337)

[2.1. SQL Statement: 3](#_Toc12109338)

[2.2. Query Result Set (Limit 5): 4](#_Toc12109341)

[3. List the manager of each department with the following information: department number, department name, the manager's employee number, last name, first name, and start and end employment dates. 4](#_Toc12109342)

[3.1. SQL Statement: 4](#_Toc12109343)

[3.2. Query Result Set (limit 5): 4](#_Toc12109347)

[4. List the department of each employee with the following information: employee number, last name, first name, and department name. 5](#_Toc12109348)

[4.1. SQL Statement: 5](#_Toc12109349)

[4.2. Query Result Set (Limit 5): 5](#_Toc12109350)

[5. List all employees whose first name is "Hercules" and last names begin with "B.". 6](#_Toc12109351)

[5.1. SQL Statement: 6](#_Toc12109353)

[5.2. Query Result Set: 6](#_Toc12109360)

[6. List all employees in the Sales department, including their employee number, last name, first name, and department name. 7](#_Toc12109361)

[6.1. SQL Statement: 7](#_Toc12109363)

[6.2. Query Result Set (Limit 5): 7](#_Toc12109364)

[7. List all employees in the Sales and Development departments, including their employee number, last name, first name, and department name. 8](#_Toc12109365)

[7.1. SQL Statement: 8](#_Toc12109373)

[order by d.dept\_name, e.last\_name, e.first\_name 8](#_Toc12109374)

[7.2. Query Result Set (limit 5): 8](#_Toc12109376)

[8. In descending order, list the frequency count of employee last names, i.e., how many employees share each last name. 9](#_Toc12109377)

[8.1 SQL Statement: 9](#_Toc12109378)

[order by frequency desc 9](#_Toc12109379)

[8.2. Query Result Set (limit 5): 9](#_Toc12109381)

[IV. Findings 9](#_Toc12109382)

[The Recommendation 10](#_Toc12109383)

[Appendix 1 11](#_Toc12109384)

[Appendix 2 12](#_Toc12109385)

[Appendix 3 14](#_Toc12109386)

## Background

Pewlett Hackard has hired me as a Data Engineer. My first task is a research project on employees from the 1980’s and 1990’s. My assignment is to design database schema’s necessary to store the provided data to perform Data Modeling, Data Engineering, and Data Analysis.

## Data Modeling

I have analyzed the CSVs files provided and used Quickdatabasediagrams.com to design an entity relational diagram, ERD of the tables necessary to store the data provided in the files formatted as CSV files. Please refer to Appendix A to review this diagram.

## Data Analysis

Data analysis was performed using a series of queries listed below: each query can be found embedded in this document as an object in [*Appendix 2*](#_Appendix_2).

1. List the following details of each employee: employee number, last name, first name, gender, and salary.

### SQL Statement:

/\* 1. List the following details of each employee:

employee number, last name, first name, gender, and salary. \*/

select e.emp\_no, e.first\_name,e.last\_name, e.gender, s.salary

from employees e, salaries s

where s.emp\_no = e.emp\_no

order by 2,3;

### Query Result Set (Limit 5):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| emp\_no | first\_name | last\_name | gender | salary |
| 69256 | Aamer | Anger | M | $40,000.00 |
| 486584 | Aamer | Armand | M | $40,000.00 |
| 237165 | Aamer | Azevdeo | F | $40,000.00 |
| 413688 | Aamer | Azuma | M | $47,553.00 |
| 281363 | Aamer | Baak | F | $62,831.00 |

## List employees who were hired in 1986.

### SQL Statement:

/\* 2. List employees who were hired in 1986 \*/

select e.emp\_no, e.last\_name, e.first\_name, e.birth\_date, e.gender, e.hire\_date

from employees e

where '1986' = right(e.hire\_date, 4) /\* list all hires from 1986 \*/

order by e.last\_name, e.first\_name



### Query Result Set (Limit 5):

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| emp\_no | last\_name | first\_name | birth\_date | gender | hire\_date |
| 29182 | Aamodt | Arumugam | 11/17/1952 | F | 1/9/1986 |
| 291658 | Aamodt | Arvind | 11/9/1953 | M | 11/20/1986 |
| 411560 | Aamodt | Basant | 7/23/1963 | F | 12/25/1986 |
| 450068 | Aamodt | Danco | 12/7/1964 | F | 10/10/1986 |
| 293563 | Aamodt | Dietrich | 8/2/1957 | F | 12/7/1986 |

## List the manager of each department with the following information: department number, department name, the manager's employee number, last name, first name, and start and end employment dates.

### SQL Statement:

/\* 3. List the manager of each department with the following information: department number, department name,

the manager's employee number, last name, first name, and start and end employment dates. \*/

select d.dept\_no, d.dept\_name, dm.emp\_no, e.last\_name, e.first\_name, dm.from\_date, dm.to\_date

from employees e, dept\_mgr dm, departments d

where e.emp\_no = dm.emp\_no

and d.dept\_no = dm.dept\_no

order by d.dept\_name, e.last\_name, e.first\_name







### Query Result Set (limit 5):

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| dept\_no | dept\_name | emp\_no | last\_name | first\_name | from\_date | to\_date |
| d009 | Customer Service | 111692 | Butterworth | Tonny | 1/1/1985 | 10/17/1988 |
| d009 | Customer Service | 111784 | Giarratana | Marjo | 10/17/1988 | 9/8/1992 |
| d009 | Customer Service | 111877 | Spinelli | Xiaobin | 9/8/1992 | 1/3/1996 |
| d009 | Customer Service | 111939 | Weedman | Yuchang | 1/3/1996 | 1/1/9999 |
| d005 | Development | 110567 | DasSarma | Leon | 4/25/1992 | 1/1/9999 |

## List the department of each employee with the following information: employee number, last name, first name, and department name.

### SQL Statement:

/\* 4. List the department of each employee with the following information: employee number, last name,

first name, and department name. \*/

select d.dept\_name, e.emp\_no, e.last\_name, e.first\_name

from employees e, depart\_emp de, departments d

where de.emp\_no = e.emp\_no

and d.dept\_no = de.dept\_no

order by d.dept\_name, e.last\_name, e.first\_name

### Query Result Set (Limit 5):

|  |  |  |  |
| --- | --- | --- | --- |
| dept\_name | emp\_no | last\_name | first\_name |
| Customer Service | 29182 | Aamodt | Arumugam |
| Customer Service | 17400 | Aamodt | Basim |
| Customer Service | 290839 | Aamodt | Guoxiang |
| Customer Service | 285421 | Aamodt | Kamran |
| Customer Service | 258030 | Aamodt | Luigi |

## List all employees whose first name is "Hercules" and last names begin with "B.".



### SQL Statement:

/\* 5. List all employees whose first name is

Hercules" and last names begin with "B." \*/

select first\_name, last\_name

from employees

where first\_name = 'Hercules'

and last\_name like 'B%'



### Query Result Set:

|  |  |
| --- | --- |
| first\_name | last\_name |
| Hercules | Benzmuller |
| Hercules | Brendel |
| Hercules | Baranowski |
| Hercules | Barreiro |
| Hercules | Baer |

## List all employees in the Sales department, including their employee number, last name, first name, and department name.



### SQL Statement:

/\* 6. List all employees in the Sales department, including their employee number, last name, first name,

and department name. \*/

select d.dept\_name, e.emp\_no, e.last\_name, e.first\_name

from employees e, depart\_emp de, departments d

where dept\_name = 'Sales'

and de.dept\_no = d.dept\_no

and e.emp\_no = de.emp\_no

order by e.last\_name, e.first\_name

### Query Result Set (Limit 5):

|  |  |  |  |
| --- | --- | --- | --- |
| dept\_name | emp\_no | last\_name | first\_name |
| Sales | 455773 | Aamodt | Aemilian |
| Sales | 107094 | Aamodt | Baoqiu |
| Sales | 203802 | Aamodt | Berry |
| Sales | 428594 | Aamodt | Breannda |
| Sales | 61219 | Aamodt | Chuanyi |

## List all employees in the Sales and Development departments, including their employee number, last name, first name, and department name.



### SQL Statement:

/\* 7. List all employees in the Sales and Development departments, including their employee number, last name,

first name, and department name. \*/

select d.dept\_name, e.emp\_no, e.last\_name, e.first\_name

from employees e, depart\_emp de, departments d

where (dept\_name = 'Sales' or dept\_name = 'Development')

and de.dept\_no = d.dept\_no

and e.emp\_no = de.emp\_no

### order by d.dept\_name, e.last\_name, e.first\_name



### Query Result Set (limit 5):

|  |  |  |  |
| --- | --- | --- | --- |
| dept\_name | emp\_no | last\_name | first\_name |
| Development | 258641 | Aamodt | Abdelkader |
| Development | 258005 | Aamodt | Adhemar |
| Development | 436560 | Aamodt | Alagu |
| Development | 15427 | Aamodt | Aluzio |
| Development | 238705 | Aamodt | Anoosh |

## **In descending order, list the frequency count of employee last names, i.e., how many employees share each last name.**

### SQL Statement:

/\* 8. In descending order, list the frequency count of employee last names, i.e., how many employees

share each last name. \*/

select last\_name, count(last\_name) as frequency

from employees

group by last\_name

order by frequency desc



### Query Result Set (limit 5):

|  |  |
| --- | --- |
| last\_name | frequency |
| Baba | 226 |
| Coorg | 223 |
| Gelosh | 223 |
| Sudbeck | 222 |
| Farris | 222 |

## Findings

After careful review there were some notable results worth mention. First thing that stood out was the names of the employees. There seem to be a lot of employees with the same name. Even many more with the same last name. These names seemed to be either foreign or plainly fictitious. Furthermore, there was only one last name that was unique to an employee. This is unusual even for a global company, such as Pewlett Hackard, an American based company with the majority of its employees in the United State. That being said the item that stands out the most is the salary break down by title. They appear to be out of sync. Review the analysis result of employee titles ordered by salary descending order.

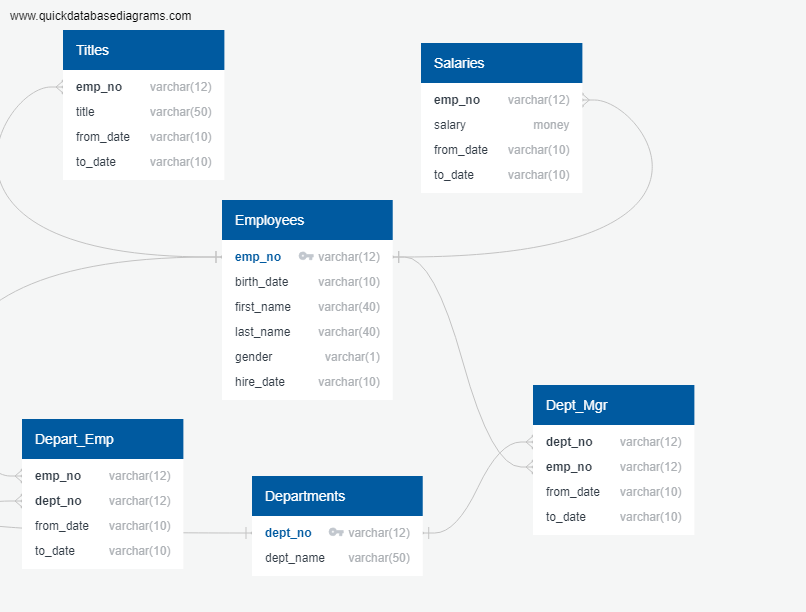
|  |  |
| --- | --- |
| **Title** | **Salary** |
| Senior Staff | $58,503.29 |
| Staff | $58,465.27 |
| Manager | $51,531.04 |
| Technique Leader | $48,580.51 |
| Engineer | $48,539.78 |
| Senior Engineer | $48,506.75 |
| Assistant Engineer | $48,493.20 |

Notice the managers earn less than the staff and senior engineers make less than the regular engineers. The assistant engineers make only $13.55 dollars less than the senior engineers. Refer to [*Appendix 3*](#_Appendix_3), to review the “Salary by Title” chart which shows each position and its related salary sorted by the highest paid to the lowest paid. This can only lead to one conclusion, which is **“we must reject the assumption this data is valid”**.

## The Recommendation

The recommendation resulting from this research is to “*find better data sources*”. First identify major problems/opportunities requiring analysis. Then determine impacts of each problem/opportunity on the business. Then figure out what questions need answers to arrive at some actionable conclusions. Once these items are resolve, then determine where valid data sources can be found that can be used to derive some relivable conclusions.

## Appendix 1



## Appendix 2

Schema for db\_design\_hwk used to analyze the provided employee data.



-- Exported from QuickDBD: https://www.quickdatatabasediagrams.com/

-- NOTE! If you have used non-SQL datatypes in your design, you will have to change these here.

-- Modify this code to update the DB schema diagram.

-- To reset the sample schema, replace everything with

-- two dots ('..' - without quotes).

CREATE TABLE Titles (

emp\_no varchar(12) NOT NULL,

title varchar(50) NOT NULL,

from\_date varchar(10) NOT NULL,

to\_date varchar(10) NOT NULL

);

CREATE TABLE Salaries (

emp\_no varchar(12) NOT NULL,

salary money NOT NULL,

from\_date varchar(10) NOT NULL,

to\_date varchar(10) NOT NULL

);

CREATE TABLE Employees (

emp\_no varchar(12) NOT NULL,

birth\_date varchar(10) NOT NULL,

first\_name varchar(40) NOT NULL,

last\_name varchar(40) NOT NULL,

gender varchar(1) NOT NULL,

hire\_date varchar(10) NOT NULL,

CONSTRAINT pk\_Employees PRIMARY KEY (

emp\_no

)

);

CREATE TABLE Dept\_Mgr (

dept\_no varchar(12) NOT NULL,

emp\_no varchar(12) NOT NULL,

from\_date varchar(10) NOT NULL,

to\_date varchar(10) NOT NULL

);

CREATE TABLE Depart\_Emp (

emp\_no varchar(12) NOT NULL,

dept\_no varchar(12) NOT NULL,

from\_date varchar(10) NOT NULL,

to\_date varchar(10) NOT NULL

);

CREATE TABLE Departments (

dept\_no varchar(12) NOT NULL,

dept\_name varchar(50) NOT NULL,

CONSTRAINT pk\_Departments PRIMARY KEY (

dept\_no

)

);

ALTER TABLE Titles ADD CONSTRAINT fk\_Titles\_emp\_no FOREIGN KEY(emp\_no)

REFERENCES Employees (emp\_no);

ALTER TABLE Salaries ADD CONSTRAINT fk\_Salaries\_emp\_no FOREIGN KEY(emp\_no)

REFERENCES Employees (emp\_no);

ALTER TABLE Dept\_Mgr ADD CONSTRAINT fk\_Dept\_Mgr\_dept\_no FOREIGN KEY(dept\_no)

REFERENCES Departments (dept\_no);

ALTER TABLE Dept\_Mgr ADD CONSTRAINT fk\_Dept\_Mgr\_emp\_no FOREIGN KEY(emp\_no)

REFERENCES Employees (emp\_no);

ALTER TABLE Depart\_Emp ADD CONSTRAINT fk\_Depart\_Emp\_emp\_no FOREIGN KEY(emp\_no)

REFERENCES Employees (emp\_no);

ALTER TABLE Depart\_Emp ADD CONSTRAINT fk\_Depart\_Emp\_dept\_no FOREIGN KEY(dept\_no)

REFERENCES Departments (dept\_no);

## Appendix 3

The following chart was generated using a Python program called “SQL\_Hwk.ipynb”. This program has been embedded as an object in this document. Double click on the icon to open the Python Notebook script.



