**Retiring Employees Over view:**

Purpose of this report is to create a table and summaries Number of Retiring Employees. All the columns and rows are selected to make sure there are no duplicates because some employees have switched titles over the years. A new table is created that includes only the most recent title of each employee.

**Challenge Summary:**

Started our challenge by creating a new repository in github and cloned it to the desktop by using git bash. Create a new folder and named it as resources. Download departments’ data, dept\_emp data, dept\_manager data, employees’ data, salaries data, and titles data from the module 7 reading and stored it into the resources folder.

Downloaded Postgres (pgadmin) to write the code.

**PART 1:**

**Employee number:**

Create a new table and select the data from the columns we are using the **SLECTE** statement and **FROM** statement is used to tell Postgres from where to get the data; in this case we are selecting data file from employees and columns emp\_no, first\_name, last\_name. An INTO statement is used for saving the data into a table named “retirement\_info.

**Code:**

SELECT emp\_no, first\_name, last\_name **–*select the columns***

INTO retirement\_info **–*Select where to save the data***

FROM employees **–*Select where to get the data***

WHERE (birth\_date BETWEEN '1925-01-01' AND '1955-12-31') **–*Selecting the birthdate between 1925 to 1955***

AND (hire\_date BETWEEN '1985-01-01' AND '1988-12-31') **– *Select the hire date between 1985 to 1988***

**Number of Employees Retiring:**

SELECT COUNT (first\_name) **--*Count the total number of employees***

FROM employees **–*Select where to get the data to count the employees***

WHERE (birth\_date BETWEEN '1925-01-01' AND '1955-12-31') **–*Selecting the birthdate between 1925 to 1955***

AND (hire\_date BETWEEN '1985-01-01' AND '1988-12-31') **– *Select the hire date between 1985 to 1988***

**Joining departments select from dept\_manager table:**

SELECT d.dept\_name, **--*Select “d” as a dept\_name.***

dm.emp\_no, **--*Select “dm” as a dept\_manager table and emp number.***

dm.from\_date, **--*Select “dm” as a dept\_manager table and from date.***

dm.to\_date **--*Select “dm” as a dept\_manager table and to date.***

FROM departments as d **--*Select “d” as departments***

INNER JOIN dept\_manager as dm ***--The INNER JOIN clause will compare both dept\_manager table and departments table, find all the paired rows and join the tables ON d.dept\_no = dm.dept\_no***

WHERE dm.to\_date = ('9999-01-01');

**Selecting current employees:**

SELECT ri.emp\_no, ***-- Use “ri” for retirement\_info table and select emp\_no.***

ri.first\_name, ***-- Use “ri” for retirement\_info table and select first name.***

ri.last\_name, ***-- Use “ri” for retirement\_info table and select last name.***

de,to\_date

INTO current\_emp --***Select where to save the data***

FROM retirement\_info as ri **–*Select where to get the data***

LEFT JOIN dept\_emp as de

ON ri.emp\_no = de.emp\_no

WHERE de.to\_date = ('9999-01-01'); ***--When this block of code is executed, a new table containing only the current employees who are eligible for retirement.***

**Employee count by department number:**

SELECT COUNT (ce.emp\_no), de.dept\_no

FROM current\_emp as ce

LEFT JOIN dept\_emp as de

ON ce.emp\_no = de.emp\_no

GROUP BY de.dept\_no

ORDER BY de.dept\_no;

**Employee list with gender and salary:**

SELECT e.emp\_no,

e.first\_name,

e.last\_name,

s.salary,

de.to\_date

INTO emp\_info

FROM employees as e

INNER JOIN salaries as s

ON (e.emp\_no = s.emp\_no)

INNER JOIN dept\_emp as de

ON (e.emp\_no = de.emp\_no)

WHERE (e.birth\_date BETWEEN '1952-01-01' AND '1955-12-31')

AND (e.hire\_date BETWEEN '1985-01-01' AND '1988-12-31')

AND (de.to\_date = '9999-01-01')

--SELECT \* FROM emp\_info;

**List of Managers in Departments:**

SELECT dm.dept\_no,

d.dept\_name,

dm.emp\_no,

ce.last\_name,

ce.first\_name,

dm.from\_date,

dm.to\_date

INTO manager\_info

FROM dept\_manager as dm

INNER JOIN departments as d

ON (dm.dept\_no = d.dept\_no)

INNER JOIN current\_emp as ce

ON (dm.emp\_no = ce.emp\_no);

**List of employees with departments:**

SELECT ce.emp\_no,

ce.first\_name,

ce.last\_name,

d.dept\_name

INTO dept\_info

FROM current\_emp as ce

INNER JOIN dept\_emp as de

ON (ce.emp\_no = de.emp\_no)

INNER JOIN departments as d

On (de.dept\_no = d.dept\_no);

**List of sales Employees:**

SELECT ce.emp\_no,

ce.first\_name,

ce.last\_name,

d.dept\_name

INTO sales\_info

FROM current\_emp as ce

INNER JOIN dept\_emp as de

On(ce.emp\_no = de.emp\_no)

INNER JOIN departments as d

ON (de.dept\_no = d.dept\_no)

WHERE d.dept\_name = 'Sales';

**List of sales and Developments:**

SELECT ce.emp\_no,

ce.first\_name,

ce.last\_name,

d.dept\_name

INTO sale\_dev

FROM current\_emp as ce

INNER JOIN dept\_emp as de

ON (ce.emp\_no = de.emp\_no)

INNER JOIN departments as d

ON(de.dept\_no =d.dept\_no)

WHERE d.dept\_name IN ('Sales', 'Development')

ORDER BY ce.emp\_no;

#### Part 2: Mentorship Eligibility:

#### This part of the challenge is to determine the employee’s mentorship eligibility. We needs Employees date of birth that falls between January 1, 1965 and December 31, 1965. Two JOINS and INNER Functions are used.

#### To calculate the number of mentorship eligibility, we include employee numbers, first and last name, title and from\_date and to\_date tables.

#### Code:

#### Number of [titles] retiring:

#### SELECT ce.emp\_no,

#### ce.first\_name,

#### ce.last\_name,

#### ti.title, --"ti for titles"

#### ti.from\_date, --"select title from date"

#### ti.to\_date

#### INTO ret\_titles

#### FROM current\_emp as ce

#### INNER JOIN titles as ti

#### ON (ce.emp\_no = ti.emp\_no)

#### ORDER BY ce.emp\_no;

#### SELECT \* FROM ret\_titles;

#### ORDER BY ret\_titels.emp\_no;

#### Partitions the data to show only most recent titles per employees:

#### SELECT emp\_no,

#### first\_name,

#### last\_name,

#### to\_date

#### title

#### INTO new\_titles

#### FROM (

#### SELECT emp\_no,

#### first\_name,

#### last\_name,

#### to\_date,

#### title, ROW\_NUMBER() OVER

#### (PARTITION BY (emp\_no)

#### ORDER BY to\_date DESC) rn

#### FROM ret\_titles

#### ) tem WHERE rn =1

#### ORDER BY emp\_no;

#### SELECT \* FROM new\_titles;

#### Counting the number of employees title:

#### SELECT COUNT(title), title

#### INTO retiring\_title

#### FROM new\_titles

#### GROUP BY title

#### ORDER BY Count DESC;

#### SELECT \* FROM retiring\_title;

#### Creating a list of employees eligible for potential mentorship program

#### SELECT e.emp\_no,

#### e.first\_name,

#### e.last\_name,

#### e.birth\_date,

#### de.from\_date,

#### de.to\_date,

#### ti.title

#### INTO mentorship

#### FROM employees as e

#### INNER JOIN dept\_emp as de

#### ON (e.emp\_no = de.emp\_no)

#### INNER JOIN titles as ti

#### ON (e.emp\_no = ti.emp\_no)

#### WHERE (de.to\_date = '9999-01-01')

#### AND(e.birth\_date BETWEEN '1965-01-01' AND '1965-12-31')

#### ORDER BY e.emp\_no;

#### SELECT \* FROM mentorship;

#### 