**Overview of the project:**

* The objective of this assignment is to:
* Determine the number of retiring employees per title,
* Identify employees who are eligible to participate in a mentorship program.
* Compile a report that summarizes your analysis and helps prepare Bobby’s manager for the “silver tsunami” as many current employees reach retirement age.

**Overview of Project**

Now that Bobby has proven his SQL chops, his manager has given both of you two more assignments: determine the number of retiring employees per title, and identify employees who are eligible to participate in a mentorship program. Then, you’ll write a report that summarizes your analysis and helps prepare Bobby’s manager for the “silver tsunami” as many current employees reach retirement age.

1. ***Deliverable 1***: The Number of Retiring Employees by Title
2. ***Deliverable 2***: The Employees Eligible for the Mentorship Program
3. ***Deliverable 3***: A written report on the employee database analysis [README.md](https://github.com/emmanuelmartinezs/Pewlett-Hackard-Analysis).

**Resources and Before Start Notes:**

* Data Source: Employee\_Database\_challenge.sql
* Data Tools: PostgreSQL, pgAdmin, Jupyter Notebook, GitHub
* Software: pgAdmin 4.26, Python 3.11, VS Code
* For more information, read the [Documentation on PostgreSQL and other data typess](https://www.postgresql.org/docs/manuals/).

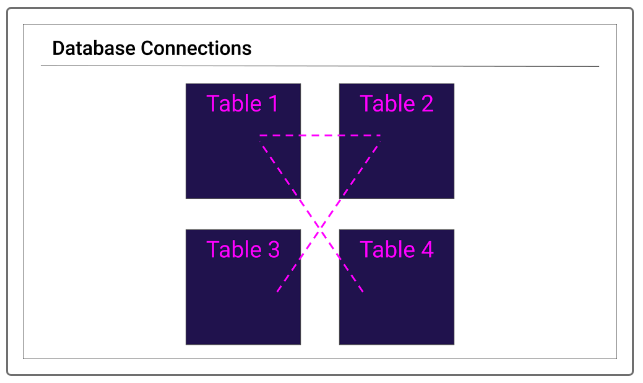
**Important concepts for this assignment:**

**Database Keys**

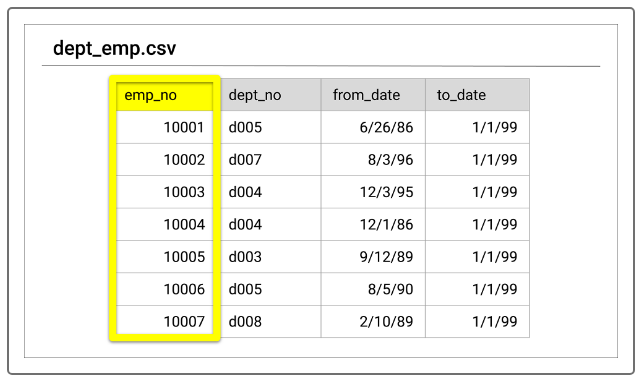
* Database keys identify records from tables and establish relationships between tables. There are numerous types of keys. For our purposes, we will focus on primary keys and foreign keys.

**Primary Keys**

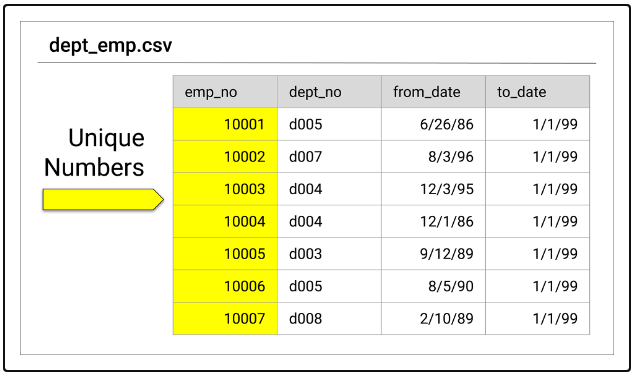
* The departments.csv file has a dept\_no column with unique identifiers for each row (one department number per department). For example, d001 will always reference the Marketing department, across other worksheets. This unique identifier is known as a primary key.
* Primary keys are an important part of database design. When a database is being created, each table added must include a primary key in the architecture. Primary keys serve as a link between these tables.

[](https://github.com/emmanuelmartinezs/Pewlett-Hackard-Analysis/blob/main/Resources/Images/s1.PNG?raw=true)

* In the graphic above, Table 1 has a primary key, or column of unique identifiers in common with Tables 2 and 4. Table 3's primary key is linked only to Table 2. These links trace the relationships between tables. There are times when we'll need to trace two or three links to get the exact data we need. In these cases, we'll pick the data we need from each table. Linking the tables together in this manner is called a join, a feature we'll get into later.
* In the second CSV file, dept\_emp.csv, the "emp\_no" column contains the primary key.

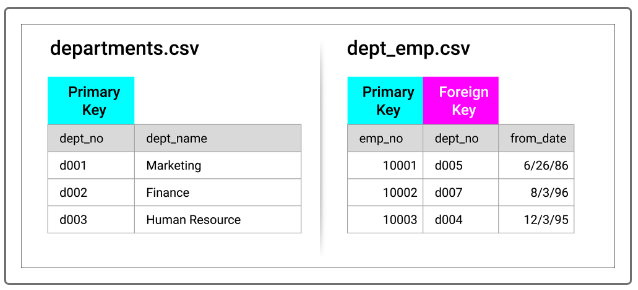
[](https://github.com/emmanuelmartinezs/Pewlett-Hackard-Analysis/blob/main/Resources/Images/s2.PNG?raw=true)

* We know this is the primary key because each number is unique. For example, the emp\_no column holds employee numbers. Each employee will have only one number, and that number won't be used for any other employee.

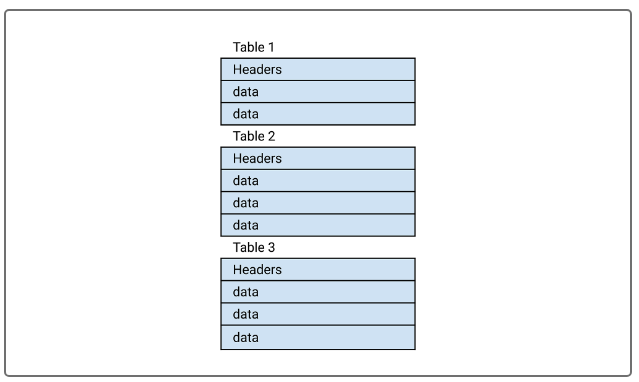
[](https://github.com/emmanuelmartinezs/Pewlett-Hackard-Analysis/blob/main/Resources/Images/s3.PNG?raw=true)

**Foreign Keys**

* Foreign keys are just as important as primary keys. While primary keys contain unique identifiers for their dataset, a **foreign key** references another dataset's primary key.
* Think about it like a phone number. You have your own number. It's your number, assigned to your phone, and unique to you. This is your primary key. Your friend also has a primary key: his or her own phone number.
* When you save your friend's number in your phone, you're creating a reference to that person, also known as a foreign key. Your phone has lots of foreign keys (such as parents, doctors offices, friends, and other family), but only one primary key.
* Likewise, when your friend saves your number in their phone, your number is now a foreign key in their phone. Saving these keys connects the devices. They show the relationship between your phone and your friend's phone.
* Compare our first two CSVs again by looking at the following image.

[](https://github.com/emmanuelmartinezs/Pewlett-Hackard-Analysis/blob/main/Resources/Images/s4.PNG?raw=true)

* In this example, dept\_no shows up in both datasets; as an identifier (or primary key) in one and as a reference (or foreign key) in the other. This demonstrates the link between employees and which department they work in.
* We could continue to look for connections between the datasets, or we could create a roadmap of the content. Our roadmap would serve as a quick reference diagramming the different datasets and their interconnections. Additionally, it could be used as a reference guide later, when we begin to create queries to access all of the data.
* **Table Structure**
* When working in Excel and Visual Basic for Applications (VBA), we're working directly with worksheets with data. In SQL, the same worksheets we have been exploring are organized into tables instead. They are similar to DataFrames in that they have headers and indexes, with data in columns and rows. Take a look at the following images.

[](https://github.com/emmanuelmartinezs/Pewlett-Hackard-Analysis/blob/main/Resources/Images/s5.PNG?raw=true)

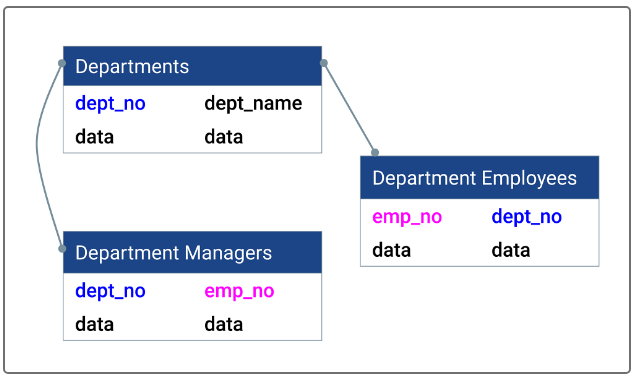
Next we'll cover how table structure comes into play when creating an entity relationship diagram.

**Entity Relationship Diagrams (ERDs)**

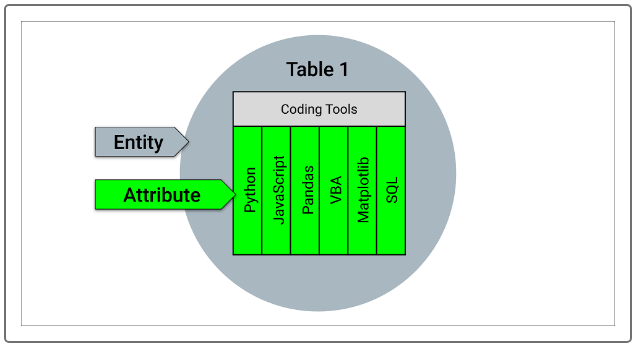
An entity relationship diagram (ERD) is a type of flowchart that highlights different tables and their relationships to each other. The ERD does not include any actual data, but it does capture the following pertinent information from each CSV file:

* Primary keys
* Foreign keys
* Data types for each column

The ERD also shows the flow of information from one table to another, as captured in the image below:

[](https://github.com/emmanuelmartinezs/Pewlett-Hackard-Analysis/blob/main/Resources/Images/s6.PNG?raw=true)

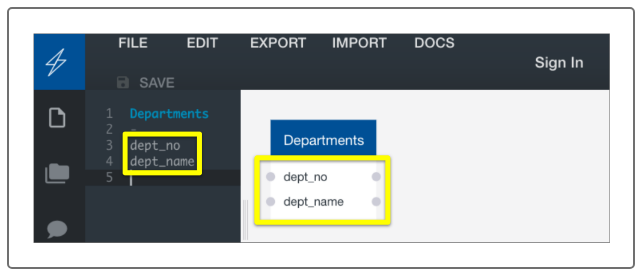
* In addition to creating new databases, ERDs are used to document existing databases. The visual representation of the tables gives a deeper understanding of the data and the database as a whole.
* When creating a diagram, we need to fully understand all of the data being inserted. Database components include tables, known as **entities**, with data, known as **attributes**.

[](https://github.com/emmanuelmartinezs/Pewlett-Hackard-Analysis/blob/main/Resources/Images/s7.PNG?raw=true)

* Data types include Booleans, integers, and varying characters (i.e., within a string).
* There are three types of ERDs: **conceptual, logical,** and **physical**. Each one builds upon the other—you need the conceptual ERD to build a logical ERD to build a physical ERD. We'll learn how to create ERDs later in this module.

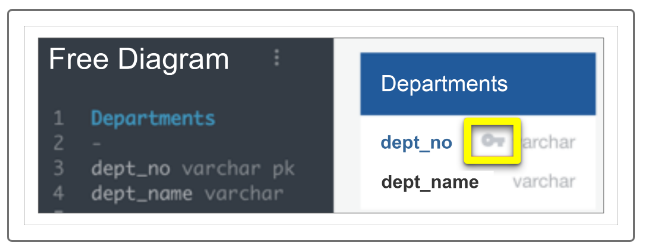
**Conceptual Diagrams**

* A conceptual diagram is an ERD in its simplest form. To create one, we only need two things: a table name and column headers.
* It's simple because we're creating just the concept of the diagram. By covering only the basics, it's easier to capture the main points. If we tried to capture everything at once (data types, location of the primary and foreign keys, etc.), we're more likely to overlook a crucial item.

[](https://github.com/emmanuelmartinezs/Pewlett-Hackard-Analysis/blob/main/Resources/Images/s10.PNG?raw=true)

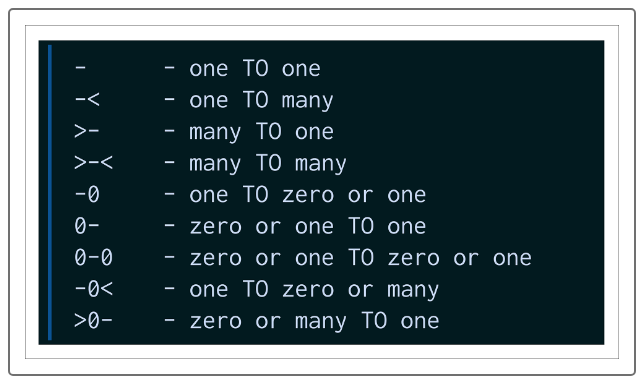
**Logical Diagrams**

* Logical diagrams contain all of the same information that a conceptual diagram does, but the table is updated to include data types and primary keys.
* Returning to the Quick DBD webpage, let's update our schema. Because we already took an initial look at the worksheet, we have already identified the primary key and know what type of data we're working with. Using the following syntax, update our Departments schema:
* Add "varchar pk" to dept\_no. Add "varchar" to dept\_name. We use varchar (Links to an external site.) in these columns because the fields contain characters of varying length. Adding "pk" in the schema next to column indicates that column as a primary key. The table updated to reflect the changes in the text editor. A key symbol appears next to the dept\_no line, indicating that it is the table's primary key, and varchar is added to indicate its type.

[](https://github.com/emmanuelmartinezs/Pewlett-Hackard-Analysis/blob/main/Resources/Images/s9.PNG?raw=true)

**Physical Diagrams**

Physical diagrams portray the physical relationship, or how the data is connected, between each table. There are several different relationships available to keep in mind when making these connections, as shown below:

[](https://github.com/emmanuelmartinezs/Pewlett-Hackard-Analysis/blob/main/Resources/Images/s8.PNG?raw=true)

**Deliverable 1: The Number of Retiring Employees by Title**

* Can be found in the the Queries folder of the project.

**Deliverable 2: The Employees Eligible for the Mentorship Program**

* Can be found in the the Queries folder of the project.

Deliverable 3: A written report on the employee database analysis

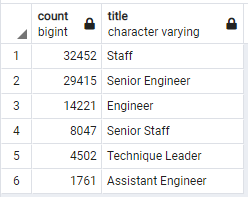
1. **Overview of the analysis**

* Explain the purpose of this analysis.:

In this deliverable, Bobby was tasked to determine the number of retiring employees per title, and identify employees who are eligible to participate in a mentorship program. Then, you’ll write a report that summarizes your analysis and helps prepare Bobby’s manager for the “silver tsunami” as many current employees reach retirement age.

1. **Results:**

* Provide a bulleted list with four major points from the two analysis deliverables. Use images as support where needed:
  + From the finding of the eligible retirees, High Percentage of the workforce could retire at any given time.
  + From the job titles of the eligible retirees, the breakdown is below.
  + 32,452 Staff
  + 29,415 Senior Engineer
  + 14,221 Engineer
  + 8,047 Senior Staff
  + 4,502 Technique Leader
  + 1,761 Assistant Engineer

[](https://github.com/emmanuelmartinezs/Pewlett-Hackard-Analysis/blob/main/Resources/Images/3.2.PNG?raw=true)

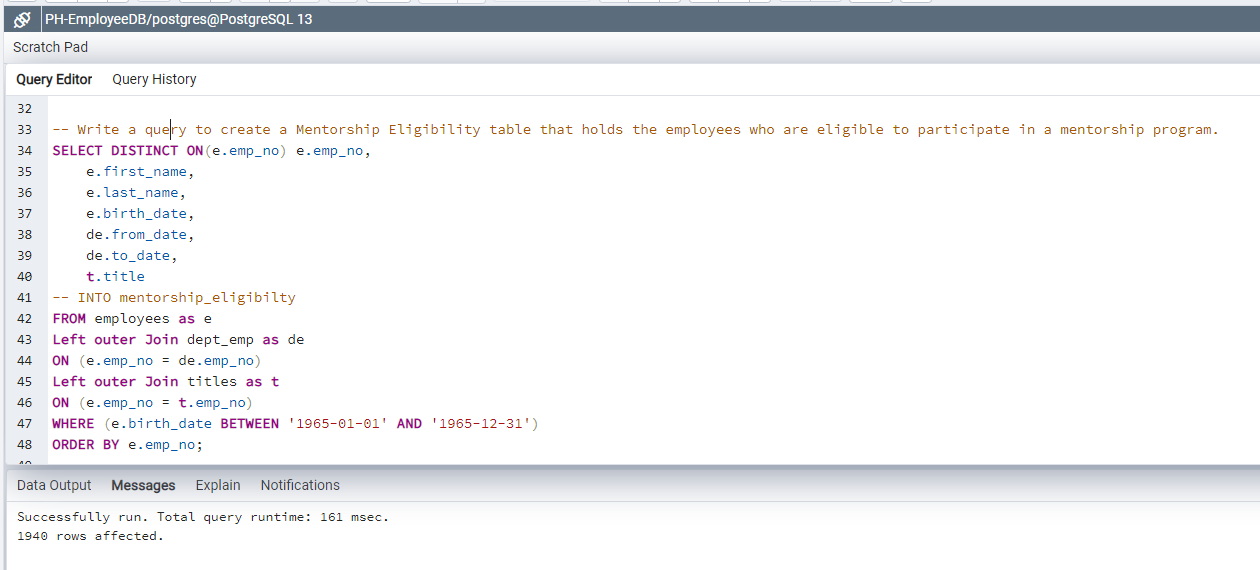
1. **Summary**

* Provide high-level responses to the following questions, then provide two additional queries or tables that may provide more insight into the upcoming "silver tsunami.":
  1. How many roles will need to be filled as the "silver tsunami" begins to make an impact?.

90,398 roles are in urgent need to be filled out as soon as the workforce starts retiring at any given time.

* 1. Are there enough qualified, retirement-ready employees in the departments to mentor the next generation of Pewlett Hackard employees?

No, we have 1,940 employees who are eligible to participate in a mentorship program.

[](https://github.com/emmanuelmartinezs/Pewlett-Hackard-Analysis/blob/main/Resources/Images/3.3.2.PNG?raw=true)