**Guided Lab - 304.6.2 - Joins and Clauses - Banking Database**

# 

# **Introduction:**

* JOIN queries allow us to walk through the relationships between two or more tables in the FROM clause.
* Joins are queries that combine the data of multiple tables based on their common ***columns (primary key and foreign key)*** and ***constraints*** to produce a combined result set.

# **Objective**

# In this lab, we will demonstrate and utilize SQL join predicates, SQL clauses, and aggregate functions.

**Objective**

After this lab, learners will have demonstrated the ability to:

* Use SQL Joins predicates
* Use SQL Clauses

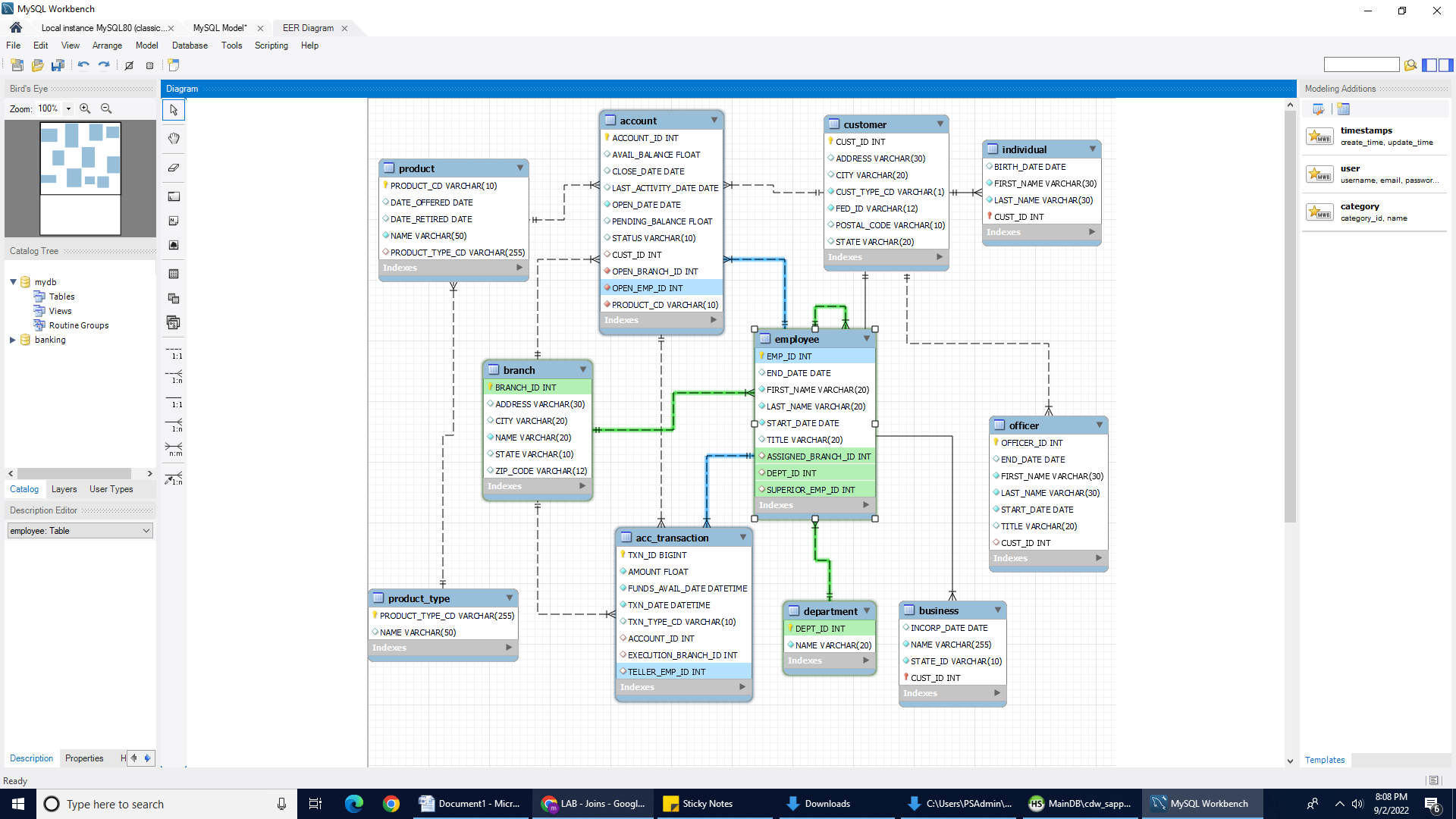
# 

# 

# **Prerequisites:**

* For this interactive lab, we will use the [Banking Database](https://drive.google.com/file/d/1aiHoX9Zc_EIzi5jrhMo4h6k45JboQmLE/view?usp=sharing).
* [Click here to download the Banking Database](https://drive.google.com/file/d/1aiHoX9Zc_EIzi5jrhMo4h6k45JboQmLE/view?usp=sharing), and set up this database in your MySQL instance.

**Schema Diagram**



## Instruction:

## Run the following solution (queries) on [Banking Database](https://drive.google.com/file/d/1aiHoX9Zc_EIzi5jrhMo4h6k45JboQmLE/view?usp=sharing) by using MySQL workbench

## 1: Problem Statement:

For each product, show the product name "Product" and the product type name "Type.

**Solution: Run the below query on MySQL:**

SELECT p.`NAME` AS "Product", pt.`NAME` AS "Type"

FROM product p INNER JOIN product\_type pt

ON p.product\_type\_cd = pt.product\_type\_cd;

## 2: Problem Statement:

For each branch, list the branch name and city, plus the last name and title of each employee who works in that branch.

**Solution: Run the below query on MySQL:**

SELECT b.`name`, b.city, e.LAST\_NAME, e.TITLE

FROM employee e INNER JOIN branch b

ON b.BRANCH\_ID = e.ASSIGNED\_BRANCH\_ID;

## 3: Problem statement:

Show a list of each unique employee title.

**Solution: Run the below query on MySQL:**

SELECT distinct TITLE FROM employee;

## 4: Problem statement:

Show the last name and title of each employee, along with the last name and title of that employee's boss.

**Solution: Run the below query on MySQL**

SELECT e.LAST\_NAME AS "Name", e.TITLE AS "Title", m.LAST\_NAME AS "Boss Name", m.TITLE AS "Boss Title"

FROM employee e LEFT JOIN employee m

ON e.SUPERIOR\_EMP\_ID = m.EMP\_ID;

## 5: Problem Statement:

For each account, show the name of the account's product, the available balance, and the customer's last name.

**Solution: Run the below query on MySQL:**

SELECT p.NAME, a.AVAIL\_BALANCE, i.LAST\_NAME FROM account a

INNER JOIN product p ON a.PRODUCT\_CD = p.PRODUCT\_CD

LEFT JOIN customer c ON a.CUST\_ID = c.CUST\_ID

LEFT JOIN individual i ON c.CUST\_ID = i.CUST\_ID;

## 6: Problem Statement:

List all account transaction details for individual customers whose last name starts with 'T'.

**Solution: Run the below query on MySQL**

SELECT ac.\*, i.LAST\_NAME FROM acc\_transaction ac

INNER JOIN account a ON ac.ACCOUNT\_ID = a.ACCOUNT\_ID

INNER JOIN customer c ON a.CUST\_ID = c.CUST\_ID

INNER JOIN individual i ON c.CUST\_ID = i.CUST\_ID

WHERE i.LAST\_NAME RLIKE "T.\*"; -- same as LIKE "T%"

**Canvas submission Instructions:** Please include the following deliverables in your submission -

* + All queries should be written and submitted in a single SQL script file.
    - Example: **<your\_name\_labname>.sql**.
  + Submit your SQL script file using the **Start** **Assignment** button in the top-right corner of the assignment page in Canvas.

**CANVAS STAFF USE ONLY: Canvas Submission Guideline:**

| **Instructions for Canvas Assignment Creation** |
| --- |
| **Assignment Name: GLAB - 304.6.2 - Joins and Clauses - Banking Database**  **Points:** **100**  **Assignment Group: Module 304 - Relational Databases and SQL - (Not Graded)**  **Display Grade As: Non-graded (This assignment does not count toward the final grade.)**.  **Complete/Incomplete**  **Do not count this assignment towards the final grade: Checked**  **Submission Types: Document File or Source Code Files**  **Allowed Attempts: Unlimited**  **Everything else is the default.** |

## 