Data Mining and Wrangling

Working with databases I

Session 7 and 8

BSDSBA 2028

11 February 2026

Class Administrative Matters

Course Deliverables

Assessed – R01, R02, R03, ICA00, ICA01, ICA02, E1

Due this Week – ICA03 (Wed EOD), E2 (Fri EOD), E1S2 (Optional – Fri EOD)

Jojie Access

Student Portal - **https://students.jojie.accesslab.aim.edu/**

Jojie Public Datasets

AIM

Common Mistakes for E1

Problem 3 – `ncr_barangay_pop`

Manila districts such as "Tondo I/II" should not be included in "City of Manila" table

Problem 5 – `get_philippines_cases`

Ensuring proper datetime format for `last_updated`

AIM

Session 7 and 8 – Working with databases I

Gameplan

Session 7

Database Fundamentals

SELECT and WHERE statements

Session 8

SQL JOINs

GROUP BY and Aggregations

AIM

Session 9 and 10 – Working with databases II

Gameplan

Session 9

Creating and Manipulating Tables

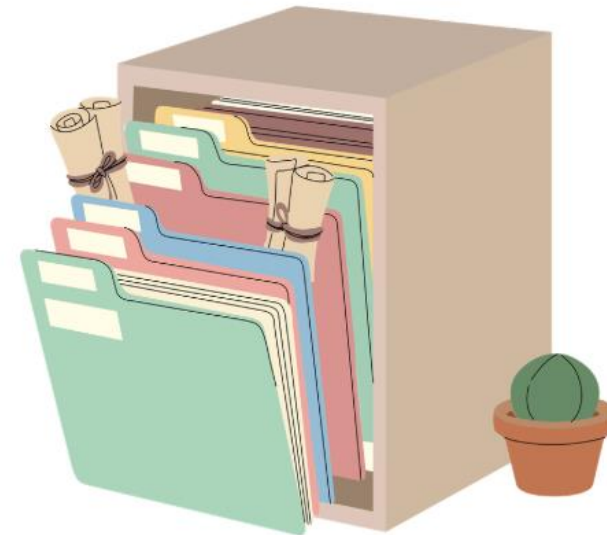Updating and Deleting Tables

Session 10

SQL: Practical Examples

AIM

Database Fundamentals

What is a database?

A **container** (usually a *file* or *set of files*) to store **organized data**

Database

AIM

Database Fundamentals
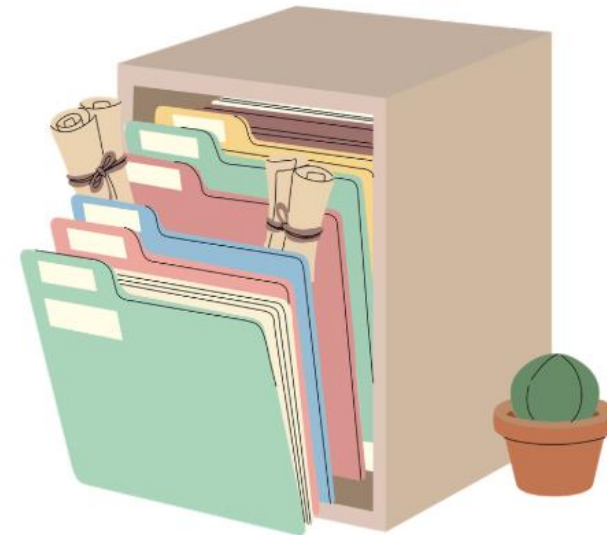
What is a database?

A **container** (usually a *file* or *set of files*) to store **organized data**

Database

"Table"

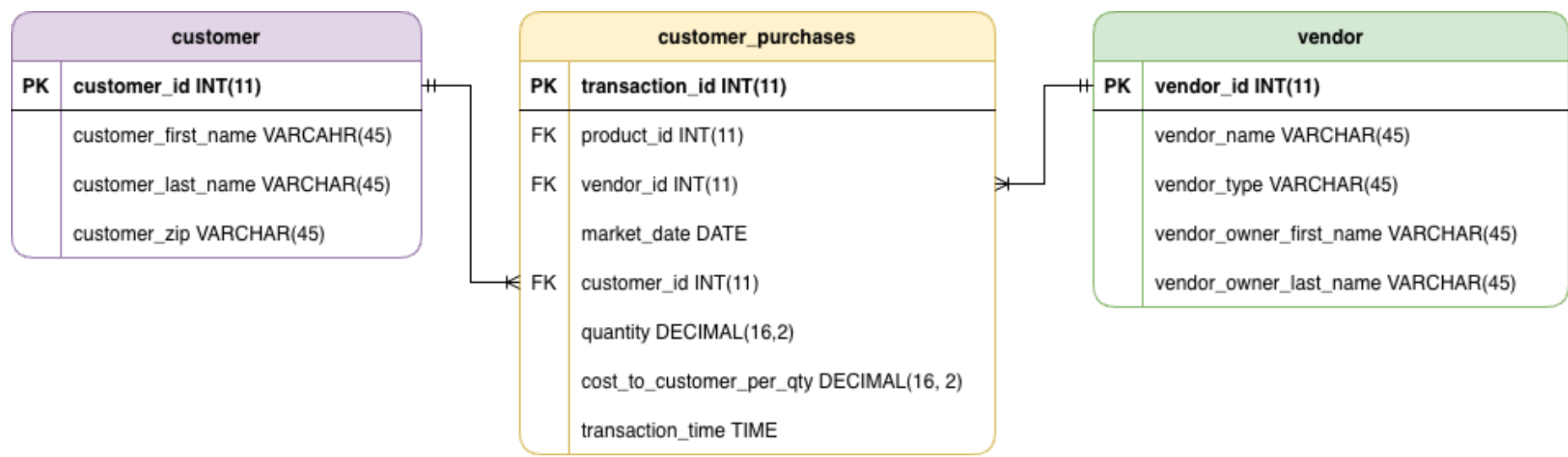A **table** is a structured list of data of a specific "*entity*".

Every table is identified by its **table name**.
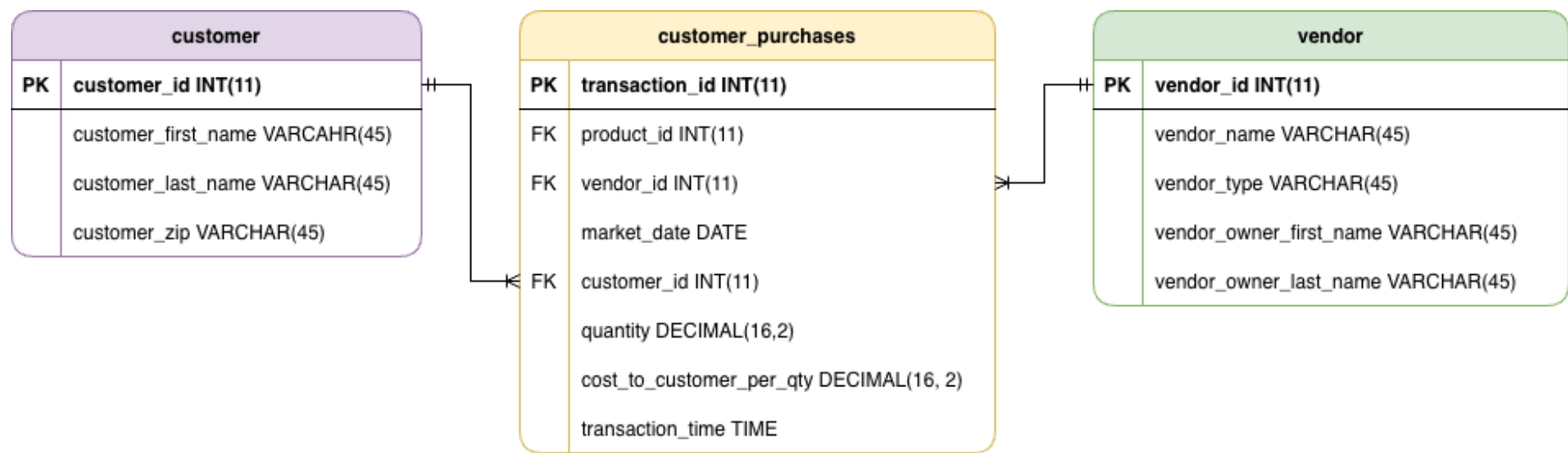
AIM

Database Fundamentals

What is a database schema?

A **database schema** stores information about the tables and relationships between them, defining the structure of the database. An *entity-relationship diagram* (ERD) is a visual representation of the database schema.

	customer
PK	customer_id INT(11)
	customer_first_name VARCAHR(45)
	customer_last_name VARCHAR(45)
	customer_zip VARCHAR(45)

	customer_purchases
PK	transaction_id INT(11)
FK	product_id INT(11)
FK	vendor_id INT(11)
	market_date DATE
FK	customer_id INT(11)
	quantity DECIMAL(16,2)
	cost_to_customer_per_qty DECIMAL(16, 2)
	transaction_time TIME

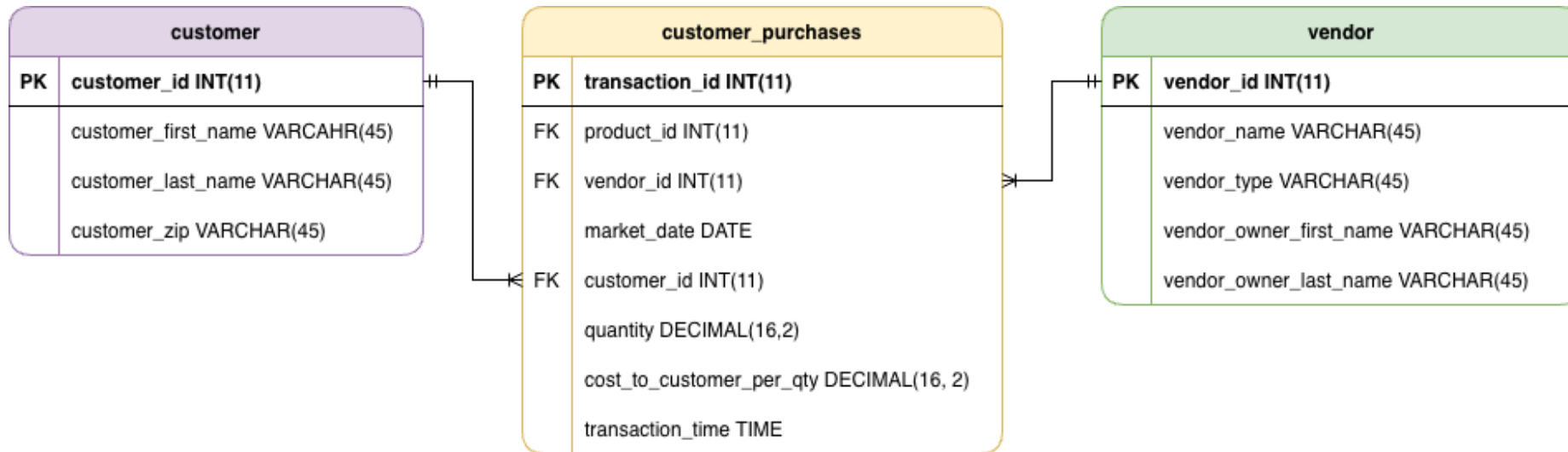|	vendor
PK	vendor_id INT(11)
	vendor_name VARCHAR(45)
	vendor_type VARCHAR(45)
	vendor_owner_first_name VARCHAR(45)
	vendor_owner_last_name VARCHAR(45)

AIM

Database Fundamentals

An example of a database: Farmer's Market DB

Columns – a particular piece of information within a table which has an associated *data type.*

AIM

Database Fundamentals

An example of a database: Farmer's Market DB

customer	
PK	customer_id INT(11)
	customer_first_name VARCAHR(45)
	customer_last_name VARCHAR(45)
	customer_zip VARCHAR(45)

customer_purchases	
PK	transaction_id INT(11)
FK	product_id INT(11)
FK	vendor_id INT(11)
	market_date DATE
FK	customer_id INT(11)
	quantity DECIMAL(16,2)
	cost_to_customer_per_qty DECIMAL(16, 2)
	transaction_time TIME

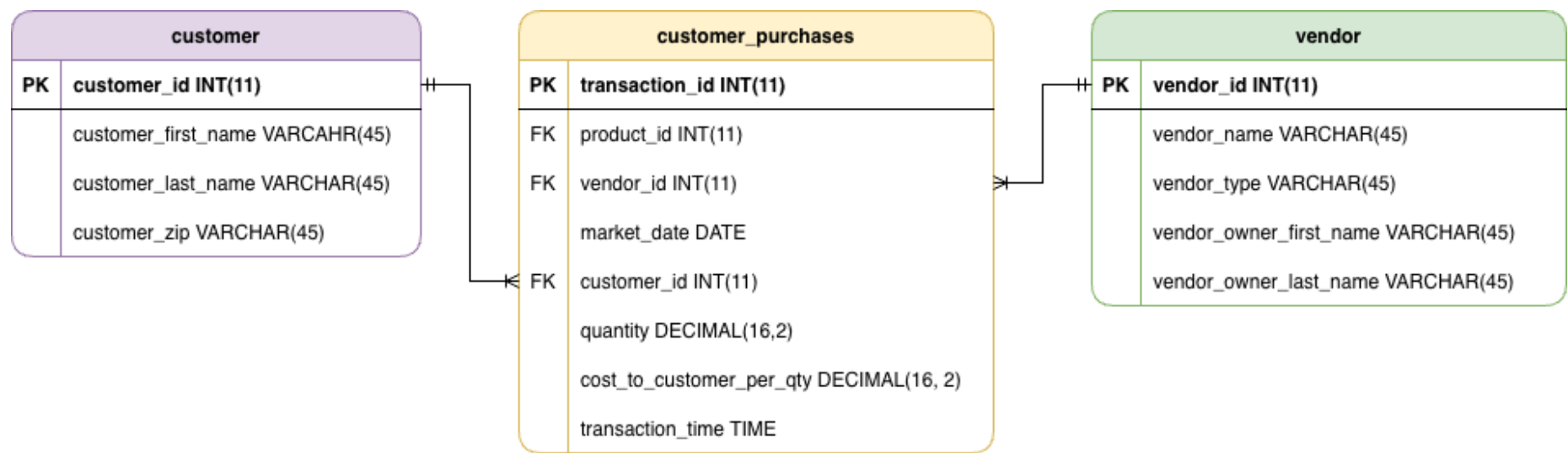| vendor	
PK	vendor_id INT(11)
	vendor_name VARCHAR(45)
	vendor_type VARCHAR(45)
	vendor_owner_first_name VARCHAR(45)
	vendor_owner_last_name VARCHAR(45)

Row – a record saved and stored in a table.

Primary Key – a column or combination of columns that uniquely identifies a row

AIM

Database Fundamentals

An example of a database: Farmer's Market DB

A **"one-to-many"** relationship denoted by   , means that a single record on one table is linked to multiple records on another table.

Database Fundamentals

What is SQL?

SQL (*S-Q-L* or *sequel)* is an abbreviation for **Structured Query Language**.

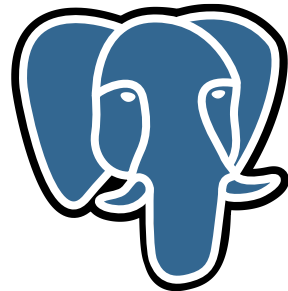SQL is a language designed specifically for communicating with databases.

AIM

Database Fundamentals

What is a DBMS (or RDBMS)?

A **Database Management Software (DBMS)** is a software used to create, manage, and interact with **databases**.

An **RDBMS** is a specialized DBMS where the stored data is organized in related tables.

Database Fundamentals

Why SQL?

- **SQL** is not a proprietary language used by a specific RDBMS. Knowing SQL allows you to interact with most database you'll encountering.

- **SQL** is readable. The statements are all made up of descriptive English words.

- **SQL** allows you to perform very complex database operations efficiently.

AIM

Database Fundamentals

How do we interact with a database in Python?

Python has a built-in package called `sqlite3`, which can be used to interact with SQL-based relational databases.

The general workflow for this is as follows:

1. Establish a connect to a database.

2. Write the **SQL query** for your desired task.

3. Execute the query.

4. Commit or fetch the result of the query.

We can also use `sqlalchemy` to create a connection with a database then use `pandas` to read the data afterwards.

AIM

Reminders

Deliverables

ICA03 – Regular Expressions (due Wednesday, Feb 11, 2026, EOD)

R04 – Working with Databases I (due Wednesday, Feb 11, 2026, EOD)

E1S2 – Data Formats (due Friday, Feb 13, 2026, EOD)

E2 – Regular Expressions (due Friday, Feb 13, 2026, EOD)

ICA04 – Regular Expressions (due Wednesday, Feb 25, 2026, EOD)

AIM